

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

**CONCEPTION ET IMPLÉMENTATION D'UNE PLATE-FORME
D'ÉVALUATION ADAPTATIVE DES APPRENTISSAGES**

MÉMOIRE PRÉSENTÉ COMME EXIGENCE PARTIELLE DE
LA MAÎTRISE EN INFORMATIQUE

PAR

KOMI SÉPÉLI SODOKÉ

MARS 2008

UNIVERSITÉ DU QUÉBEC À MONTRÉAL
Service des bibliothèques

Avertissement

La diffusion de ce mémoire se fait dans le respect des droits de son auteur, qui a signé le formulaire *Autorisation de reproduire et de diffuser un travail de recherche de cycles supérieurs* (SDU-522 – Rév.01-2006). Cette autorisation stipule que «conformément à l'article 11 du Règlement no 8 des études de cycles supérieurs, [l'auteur] concède à l'Université du Québec à Montréal une licence non exclusive d'utilisation et de publication de la totalité ou d'une partie importante de [son] travail de recherche pour des fins pédagogiques et non commerciales. Plus précisément, [l'auteur] autorise l'Université du Québec à Montréal à reproduire, diffuser, prêter, distribuer ou vendre des copies de [son] travail de recherche à des fins non commerciales sur quelque support que ce soit, y compris l'Internet. Cette licence et cette autorisation n'entraînent pas une renonciation de [la] part [de l'auteur] à [ses] droits moraux ni à [ses] droits de propriété intellectuelle. Sauf entente contraire, [l'auteur] conserve la liberté de diffuser et de commercialiser ou non ce travail dont [il] possède un exemplaire.»

Remerciements

Au terme de ce travail, j'aimerais exprimer ma profonde reconnaissance à toutes les personnes qui m'ont aidé, orienté et soutenu durant mes deux années de maîtrise.

Je remercie tout particulièrement mon directeur de mémoire, monsieur Roger Nkambou, professeur au département d'informatique à l'UQAM et mon codirecteur de mémoire, Monsieur Gilles Raïche, professeur au département d'éducation et de pédagogie de l'UQAM. Leur confiance, leur patience et leurs conseils avisés ont permis la réalisation de ce travail. Je leur dois beaucoup de reconnaissance pour leur disponibilité et leur rigueur académique.

Ce mémoire s'inscrit dans les activités du CAMRI (Centre sur les Applications des Modèles de Réponse aux Items) en collaboration avec le laboratoire GDAC (gestion, diffusion et d'acquisition des connaissances). Ces laboratoires ont fourni le financement qui a permis de mener à terme ce travail. Ils ont également mis à ma disposition d'autres chercheurs et collègues dont les suggestions et commentaires ont positivement influencé notre recherche. Je pense particulièrement au Professeur Martin Riopel et aux autres collègues de ces laboratoires à savoir : David Magis, Mohamed Gaha, Daniel Dubois, etc.

Je remercie tous les membres de ma famille pour leurs encouragements, leur soutien psychologique et leur confiance, en particulier mon père Kodjo SODOKE et ma mère HOR Kafui SODOKE. Leurs conseils et leur soutien constants durant ces deux années m'ont motivé à entreprendre mes études de maîtrise.

À tous les professeurs qui me feront l'honneur de contribuer à l'évaluation de ce travail j'adresse mes gratitude.

TABLE DES MATIÈRES

Remerciements.....	i
Liste des figures.....	vii
Liste des tableaux.....	vii
Liste des abréviations.....	viii
RÉSUMÉ.....	ix

INTRODUCTION

1.1 Contexte	1
1.2 Problématique	3
1.3 Objectifs de recherche et de développement.....	5
1.4 Plan du mémoire	7

CHAPITRE II

ÉTAT DE L'ART SUR L'ADAPTATION DES ÉVALUATIONS

2.1 Les limites des tests classiques	9
2.2 Propositions d'adaptation des évaluations en éducation.....	11
2.2.1 Test de Binet	13
2.2.2 Test à deux étapes	14
2.2.3. Test pyramidal.....	14
2.3 Approches d'adaptation proposées en informatique	16
2.3.1 Les systèmes hypermédia adaptatifs	16
2.3.2 Les systèmes d'adaptation selon le profil culturel	17
2.3.3 Les AIWBES.....	18
2.4 Conclusion du chapitre.....	19

CHAPITRE III

LES SYSTÈMES TUTORIELS INTELLIGENTS

3.1 Définition et fonctionnement	20
3.1.1 Définition et exemple.....	20
3.1.2 Principe de fonctionnement des STI	22
3.2 Architecture d'implémentation des STI.....	22
3.2.1 Le modèle de l'apprenant.....	22
3.2.2 Le modèle du domaine	24
3.3 Conclusion du chapitre.....	24

CHAPITRE IV

LES MODÈLES DE CALCUL ET DE REPRÉSENTATION

4.1 La Théorie de la Réponse à l'Item (TRI)	26
4.1.1 Définition	26
4.1.2 Les fondements de la TRI	27
4.2 La modélisation de la TRI utilisée	28
4.2.1 Les paramètres d'items dans le modèle 4PL.....	28
4.2.2 La courbe caractéristique d'item.....	29
4.2.3 Équation du modèle.....	30
4.3 Stratégies d'estimation des valeurs des paramètres	31
4.3.1 Estimation du paramètre de sujet (θ).....	31
4.3.2 Estimation des paramètres des items	34
4.4 Autres indices de la TRI à utiliser.....	35
4.4.1 L'information	35
4.4.2 L'indice d'ajustement L_z	36
4.5 Conclusion sur la TRI	38
4.6 Les réseaux bayésiens (RB)	39
4.6.1 Définition	39
4.6.2 Exemple simple de réseau bayésien.....	39
4.6.3 Utilisation des RB dans PersonFit	41

4.7 La complémentarité entre TRI et les RB dans PersonFit	42
4.8 Des approches intéressantes non utilisées.....	43
4.9 Conclusion du chapitre.....	43

CHAPITRE V

PRINCIPES DIRECTEURS ET FONCTIONNALITÉS DE PERSONFIT

5.1 Objectifs et fonctionnalités	44
5.2 Les principes directeurs.....	47
5.2.1 Les normes du E-learning utilisées dans PersonFit.....	47
5.2.2 Problèmes de confiance dans les métadonnées des normes actuelles.....	50
5.2.3 La plate-forme libre retenue pour la vulgarisation de PersonFit.....	52
5.3 Conclusion du chapitre.....	53

CHAPITRE VI

LES COMPOSANTS DE PERSONFIT ET LEURS INTERACTIONS

6.1 Les différents modules dans PersonFit	54
6.1.1 Modélisation des connaissances du domaine	55
6.1.2 La modélisation de l'apprenant.....	56
6.1.3 L'agent tuteur	61
6.2 Le QuestionRecommender.....	61
6.3 Conclusion du chapitre.....	69

CHAPITRE VII

DIAGNOSTIC COGNITIF ET DÉTECTION DE JEU DANS PERSONFIT

7.1 Le diagnostic cognitif de l'apprenant.....	70
7.1.1 Le concept de diagnostic cognitif dans les STI.....	70
7.1.2 Implémentation du diagnostic cognitif dans PersonFit.....	72
7.2 La détection de jeu dans PersonFit	77
7.2.1 Le problème de jeu dans les STI.....	77

7.2.2 Les approches de solutions au jeu dans les STI	79
7.2.3 L'indice Lz	80
7.3 Conclusion du chapitre.....	81
CHAPITRE VIII	
IMPLÉMENTATION DE PERSONFIT	82
8.1 Les langages de programmation et systèmes utilisés.....	82
8.2 Le format et la structure des items	85
8.2.1 Le format de stockage des items et paramètres d'item.....	85
8.2.2 La structure de données pour la représentation des items.....	86
8.2.3 L'interface de présentation des items.....	91
8.3 Organisation de la banque d'items.....	92
8.4 Intégration et interactions avec Moodle.....	95
8.5 Conclusion du chapitre.....	100
CHAPITRE IX	
LES RÉSULTATS OBTENUS	101
9.1 Résultat de l'utilisation des tests adaptatifs	101
9.2 Résultat de la détection de jeu.....	104
9.3 Résultat du diagnostic cognitif des apprenants	108
9.4 Conclusion du chapitre.....	108
CONCLUSION.....	109
10.1 Synthèse	109
10.2 Contributions.....	110
10.3 Améliorations et perspectives	113
Bibliographie.....	116

Liste des figures

Figure		Page
3.1	Représentation du modèle du type recouvrement, avec une variante incorporant les connaissances erronées de l'apprenant	23
4.1	CCI de deux items	30
4.2	Un RB représentant les dépendances entre cinq variables	40
6.1	Architecture simplifiée des modules de PersonFit et leurs interactions	55
6.2	Modélisation des concepts pour un cours de base de données	56
6.3	Structure d'un RB couvrant les concepts d'un cours de base de données	59
6.4	Organigramme simplifié de l'algorithme de test adaptatif	62
7.1	Boucle de diagnostic cognitif et remédiation dans PersonFit	72
7.2	Extrait du modèle des connaissances du domaine	73
7.3	Estimation de la valeur du niveau d'habileté des pré-requis	74
8.1	Structure d'un objet d'évaluation (OE)	86
8.2	Écran d'ajout de métadonnées à un nouvel item	88
8.3	Écran d'ajout d'un item à choix multiple	89
8.4	Interface client de présentation des items dans PersonFit	91
8.5	Interface de présentation du contenu de la banque locale d'items	93
8.6	Interface Moodle d'affichage de la banque d'items sur le serveur	94
8.7	Architecture d'interaction entre des composants de PersonFit et Moodle	96
9.1	Évolution du calcul du niveau d'habileté pour les valeurs de θ_i	103
9.2	Résultat de la détection lorsque les réponses sont données au hasard	106
9.3	Résultat de la détection lorsque des réponses incorrectes sont données	106

Liste des tableaux

Tableau		Page
2.1	Déroulement d'un test classique (Raïche, 2000)	12
2.2	Déroulement d'un test de Binet	13
2.3	Déroulement d'un test à deux étapes	14
2.4	Déroulement d'un test pyramidal	15
5.1	Quelques éléments du noyau simple de Dublin	48
6.1	Différents variables associées au concept de traitement des requêtes	59
6.2	Table de probabilité correspondant à $p(A B, C)$	60
6.3	Exemple fictif d'une table d'information	66
6.4	Déroulement d'un test adaptatif basé sur la TRI	69
7.1	Exemple simplifié de fiche de diagnostic cognitif d'un apprenant	76
8.1	Intervention des différents systèmes dans le déroulement d'un test adaptatif	84
8.2	Exemple de fichier de paramètre d'item	85
8.3	Option de partage des items et valeurs associées	85
8.4	Exemple de table d'information pour simulation	97

Liste des abréviations

EAP	Expected A Posteriori
EIAH	Environnement Informatique pour l'Apprentissage Humain
IA	Intelligence Artificielle
MLE	Maximum Likelihood Estimation
MAP	Maximum A Posteriori
RB	Réseaux Bayésiens
STI	Systèmes Tutoriels Intelligents
TRI	Théorie de la Réponse à l'Item
XML	eXtensible Markup Language

RÉSUMÉ

Des avancées importantes ont été réalisées au cours de ces dernières décennies dans le domaine du *E-learning*. Ceci permet par exemple l'interopérabilité des plates-formes et l'adaptation du déroulement des apprentissages à l'apprenant. Mais certains composants du *E-learning* comme l'évaluation en ligne des apprentissages, sont encore en phase de développement. Comme conséquence de ce retard, dans la plupart des plates-formes de *E-learning*, les évaluations sont sous le format classique des tests malgré leurs limites et problèmes de précision. Or, en mettant à profit des techniques d'intelligence artificielle (IA), des théories en psychométrie et les normes actuelles dans le domaine du *E-learning*, il est possible d'intégrer des fonctionnalités permettant d'administrer des évaluations adaptatives et plus informatives à ces plates-formes. Les travaux présentés dans ce mémoire se situent dans ce contexte.

Le mémoire présentera des algorithmes et des stratégies permettant l'adaptation des évaluations selon le niveau des habiletés cognitives des apprenants. Les résultats de ces évaluations serviront à faire un diagnostic cognitif sur les apprentissages de ces apprenants. Pour ce faire, nous devons nous assurer de la véracité des réponses fournies par les apprenants. Un mécanisme de détection de patrons de réponses inappropriées sera donc implémenté. Cette dernière fonctionnalité et le diagnostic cognitif seront présentés de façon sommaire. La plate-forme développée qui intègre toutes ses fonctionnalités est nommée **PersonFit**. Elle sera présentée ainsi que des stratégies permettant son intégration dans la plate-forme de *E-learning* Moodle. Finalement, une présentation et discussion sur les résultats d'implémentation permettront de juger de la pertinence et de l'efficacité du travail effectué.

MOTS-CLÉS: *E-learning*, Systèmes Tutoriels Intelligents, Théorie de la Réponse à l'Item, *E-testing*, Modèle de l'apprenant, Adaptation à l'apprenant, Moodle, PersonFit

CHAPITRE I

INTRODUCTION

1.1 Contexte

Le *E-learning*¹ concerne l'utilisation de l'ordinateur et des technologies de réseau informatique pour créer, fournir, contrôler et soutenir l'apprentissage; ceci habituellement de façon indépendante d'un endroit spécifique ou du temps (Brace et Nejd, 2004). Il s'agit actuellement d'une industrie florissante avec une utilisation croissante dans plusieurs domaines comme la formation industrielle, l'éducation post-secondaire, l'armée, le gouvernement, etc. L'intérêt que ces divers organismes portent au *E-learning* vient des nombreux avantages qu'il offre pour les apprenants ainsi que pour les formateurs, les universités et les entreprises, etc. Ceci s'explique par le fait qu'il permet à un apprenant d'avoir accès à des ressources d'apprentissage diversifiées, de bonnes qualités (images, audio, vidéo, simulations, etc.) avec peu de contraintes quant au lieu de réception. Le *E-learning* offre également à l'apprenant la possibilité d'adopter un horaire d'apprentissage facilitant la conciliation entre la formation et d'autres activités. Pour les formateurs, en plus de la possibilité de mettre à la disposition de leurs apprenants des ressources pédagogiques de bonne qualité et beaucoup plus riches en format de présentation, le *E-learning* permet de joindre plus

¹ Ce terme sera utilisé dans ce mémoire pour faire référence au téléapprentissage et E-formation

facilement des étudiants dans un environnement non supervisé et géographiquement éloigné grâce à des plates-formes informatiques de plus en plus puissantes et de faible coût, voir gratuites. Finalement, pour les entreprises, les universités et les institutions offrant des formations telles que le gouvernement ou l'armée, le *E-learning* permet de réduire les coûts de formation d'employés dispersés géographiquement et d'augmenter leur capital de savoir et de connaissances via diverses collaborations.

Des progrès considérables ont été réalisés au cours de ces dernières décennies dans le domaine du *E-learning*. Nous pouvons citer entre autres, l'interopérabilité des plates-formes et des ressources via l'adoption et l'utilisation de normes, l'intégration du design pédagogique aux plates-formes de *E-learning*, l'intégration de diverses composantes du processus d'apprentissage et l'adaptation selon le profil de l'apprenant.

L'évaluation est une des composants de l'apprentissage comme l'illustre cette définition de (Linn et Gronlund, 1995) « l'évaluation est un terme général qui inclut un ensemble complet de procédures utilisées pour obtenir de l'information à propos de l'apprentissage d'un étudiant et porter un jugement de valeur concernant la progression de celui-ci ». De part cette définition, l'évaluation joue un rôle important dans le processus d'apprentissage. Malgré ce rôle déterminant de l'évaluation, il y a lieu de noter que, dans la majorité des plates-formes de *E-learning*, les fonctionnalités associées à l'évaluation en ligne des apprentissages (*E-testing*) sont très limitées ou se trouvent en phase de développement. Au nombre des manifestations de ce retard du *E-testing*, nous pouvons noter que dans les plates-formes actuelles de *E-learning*, les évaluations des apprentissages sont toujours sous le format « classique » des tests, c'est-à-dire sur un formulaire Web identique à la version papier du test. Également, les fonctionnalités de conception et de présentation des questions d'évaluation sont généralement assez limitées. En mettant à profit la puissance de calcul que nous offrent actuellement les ordinateurs, des théories bien connues en psychométrie ainsi

que des normes actuelles dans le domaine du *E-learning*, il serait possible de trouver des solutions efficaces aux manques observés. Ceci permettra d'étendre les fonctionnalités des plates-formes de *E-learning* existantes pour offrir par exemple des évaluations adaptatives. Ces dernières sont des évaluations sur mesure dans lesquelles les tâches et questions administrées varient selon l'apprenant et selon différents critères. Ceci permet d'avoir des évaluations plus informatives et combler certaines limites des tests classiques.

Le présent travail porte sur l'élaboration d'une plate-forme permettant d'apporter des solutions à certains des retards précédemment cités. Il vise à mieux tirer profit des avantages offerts par le *E-testing* comme la réduction des coûts liés à l'administration des tests, la production de rapports informatisés en moins de temps et la conservation à long terme des résultats afin de faire des analyses statistiques et des diagnostics. Dans la prochaine section, nous présentons en détail la problématique de notre recherche.

1.2 Problématique

Dans le contexte de l'évaluation, l'adaptation consiste à produire une suite de questions qui tiennent compte de différents paramètres associés à l'apprenant (par exemple, son niveau d'habileté) et des données associées au déroulement de la session d'évaluation (temps de réponse, état émotionnel, etc.). L'adaptation d'une évaluation peut donc se faire en fonction du contenu, du format de présentation, des ressources utilisées, etc.

Actuellement, les besoins dans le domaine de l'évaluation et l'existence d'un environnement favorable nous poussent à envisager l'adaptation des évaluations en ligne et justifient la pertinence de tous les efforts menés dans ce sens. Au nombre de ceux-ci, nous pouvons citer :

- La nécessité d'évaluer des apprenants géographiquement dispersés, ayant différents attentes, cultures et niveaux d'habileté sur un même concept.
- L'existence de normes et standards dans le *E-learning* qui, utilisés adéquatement, pourront permettre l'échange et l'interopérabilité des banques de questions.
- La puissance de calcul que nous offrent actuellement les ordinateurs; ceci permet de faire en ligne des calculs assez avancés et en temps réel.

Mais l'adaptation des évaluations revêt un caractère particulier. En effet, elle doit prendre en compte les objectifs visés par l'évaluation et soutenir les décisions à prendre quant aux résultats obtenus. Traditionnellement, les résultats des évaluations permettent de porter un jugement sur les apprentissages, de prendre une décision quant à une sanction, un classement ou un diagnostic. Ces fonctions doivent être prises en compte dans l'adaptation du *E-testing* en prenant en considération les critiques et avantages qu'offrent ce dernier. En effet, malgré les nombreux avantages qu'offre le *E-testing*, des critiques portent sur son efficacité réelle à livrer des résultats valides et non biaisés en regard du traitement équitable des candidats, dans le respect des normes de pratique en matière de psychométrie et d'évaluation. Ces critiques se fondent sur des constats parmi lesquels la possibilité de distraction du répondant faisant l'évaluation en mode non supervisé; les incompatibilités technologiques du poste client du répondant qui peuvent affecter l'affichage des items de l'évaluation, la gestion de la durée du test ou le transfert sécurisé des résultats; le biais éventuel de familiarité avec les ordinateurs et la question de l'équivalence avec les épreuves *papier-crayon*. Ces différentes raisons nous poussent à ne pas envisager l'utilisation des résultats des évaluations adaptatives dans un but certificatif. L'emphase serait plutôt mise sur les *évaluations diagnostiques*. Ces évaluations ont pour but de déterminer les sources des mauvais scores obtenus par l'apprenant.

Le diagnostic est une fonction importante de l'évaluation dont on a encore peu fait l'utilisation dans les plates-formes de *E-learning*. Étant donné que les raisons

précitées nous motivent à l'utilisation des évaluations dans un but diagnostic, il faudrait mettre en œuvre des stratégies pour s'assurer que les réponses fournies par un apprenant peuvent être considérées comme des manifestations de ses connaissances et ne sont pas volontairement biaisées par ce dernier.

Au regard de cette problématique, des objectifs de recherche et de développement ont été définis et seront présentés dans la prochaine section.

1.3 Objectifs de recherche et de développement

Le travail effectué dans le cadre de ce mémoire vise le développement d'une plate-forme offrant des fonctionnalités d'adaptation des tests selon le niveau des habiletés cognitives des apprenants. La plate-forme développée devra être intégrée dans une plate-forme de *E-learning* existante pour en étendre les fonctionnalités ainsi que bénéficier de sa clientèle déjà existante pour la mise à l'essai. Les résultats des tests adaptatifs serviront à faire un diagnostic cognitif (section 7.1.1) des apprentissages des apprenants évalués. Mais avant de procéder au diagnostic cognitif, des mécanismes de détection de patrons de réponses inappropriés seront utilisés. À part ces principaux objectifs de développement, la plate-forme offrira des mécanismes permettant la création, l'importation, l'exportation et le partage de questions entre les participants au projet conformément aux normes et standards actuels dans le domaine du *E-learning*.

Par analogie à un humain, les objectifs de la plate-forme à développer seraient de reproduire le comportement qu'aurait un tuteur expérimenté qui prendrait des décisions sur les questions à poser à un répondant afin d'estimer le plus précisément possible son niveau d'habileté et de diagnostiquer d'éventuels problèmes d'apprentissage. Il semble évident que ce tuteur apprendrait peu sur l'apprenant en lui posant des questions trop difficiles, pour lesquelles il n'obtiendrait que de mauvaises

réponses, ou trop faciles, pour lesquelles il n'obtiendrait que de bonnes réponses. Au contraire, le tuteur devrait tenter, selon les informations préalables dont il dispose sur les connaissances de l'apprenant et selon le déroulement de la session d'évaluation, d'adapter le niveau de difficulté des questions administrées. Les résultats des évaluations des apprenants serviraient par la suite au tuteur pour établir un diagnostic cognitif sur les apprentissages de l'apprenant.

À notre connaissance, il n'existe pas de plate-forme dont le développement est effectué dans cette perspective et ceci constituerait donc une originalité du travail effectué dans ce mémoire.

Ces objectifs de développement serviront de base aux objectifs de recherche. Ils impliquent qu'il nous faudra tout d'abord effectuer une recherche de modélisations adéquates pour réaliser le développement voulu. Les modélisations recherchées doivent permettre de faire l'initialisation du profil de l'apprenant au début du test en se basant sur les informations connues ou qui peuvent être inférées sur ce dernier. Elles doivent également soutenir des mécanismes visant à sélectionner les items à administrer à l'apprenant afin d'obtenir le plus précisément possible son niveau d'habileté, contrôler de façon globale la séquence des questions propre à chaque apprenant et définir les règles d'arrêt de l'évaluation. Étant donné que la plate-forme pourrait être amenée à administrer différentes questions aux apprenants dans un même test, les modélisations recherchées devraient donc offrir des métriques pour évaluer et comparer de façon objective des apprenants ayant reçu différentes questions durant un test. Finalement, en utilisant le modèle, il doit être possible de détecter chez les apprenants des tentatives de jeu ou de sous performance volontaire afin de garantir la fiabilité des résultats.

La plate-forme à développer et qui va intégrer les fonctionnalités précitées porte le nom de **PersonFit** et en sera à sa version 1.0. Les bases théoriques des éventuelles

futures versions de PersonFit seront également présentées. La prochaine section, porte sur le plan adopté pour ce mémoire afin d'atteindre les objectifs visés.

1.4 Plan du mémoire

Nous avons organisé la présentation de ce mémoire en dix chapitres dont le premier est l'introduction. De façon générale, les premiers chapitres du mémoire abordent les différents modèles et fondements théoriques qui serviront de base aux développements. Les chapitres subséquents portent principalement sur l'implémentation de PersonFit, les algorithmes utilisés, les choix technologiques effectués et son intégration dans la plate-forme Moodle.

Le chapitre II présente un état de l'art sur l'adaptation des évaluations. Les faiblesses des évaluations classiques seront présentées de même que les solutions et avantages offerts par une approche adaptative.

Le chapitre III décrit les Systèmes Tuteurs Intelligents (STI) car leur architecture servira de modèle pour la conception et l'implémentation de PersonFit.

Dans le chapitre IV, les modélisations utilisées seront présentées ainsi que leurs interactions. Nous retrouverons ensuite la pertinence de leurs utilisations pour atteindre les objectifs visés.

Le chapitre V marque le début des chapitres portant sur l'implémentation de PersonFit. Il détaille les fonctionnalités de PersonFit de même que les principes qui serviront de fondements à son implémentation.

Le chapitre VI offre une description détaillée du fonctionnement de PersonFit ainsi que de l'intégration des différentes modélisations et normes. L'algorithme général de déroulement de test adaptatif utilisé sera présenté et expliqué en détails.

Le chapitre VII a été écrit pour alléger le chapitre VI. Il présentera séparément et de façon succincte deux fonctionnalités connexes de PersonFit à savoir : la détection de patrons de réponses inappropriées lors des évaluations et le diagnostic cognitif des apprenants.

Le chapitre VIII portera sur l'implémentation de PersonFit. Il portera sur les choix technologiques effectués pour les implémentations, la présentation du fonctionnement de la plate-forme avec quelques captures d'écran à l'appui.

Le chapitre IX permettra de présenter les résultats et discussion sur le travail effectué.

Le dernier chapitre est la conclusion. Il fera la synthèse de notre travail tout en relevant les contributions. Des perspectives ainsi que les futurs axes d'améliorations ou d'explorations seront proposées.

CHAPITRE II

ÉTAT DE L'ART SUR L'ADAPTATION DES ÉVALUATIONS

Ce chapitre présente un état de l'art sur l'adaptation des évaluations. Les limites des tests classiques y sont présentées de même que les avantages offerts par une approche adaptative. L'adaptation constitue un vaste sujet de recherche en *E-learning*. Nous ferons un survol de certains de ces aspects ainsi que de quelques approches d'adaptation des tests proposées en sciences de l'éducation.

2.1 Les limites des tests classiques

Dans les tests classiques, une même modalité d'évaluation est utilisée pour tous les apprenants. Lorsqu'il s'agit d'un examen, cette modalité d'évaluation est constituée d'un nombre fixe de questions ayant différents degrés de difficulté; sa durée d'administration est, elle aussi, fixe et prédéterminée. La note, qui correspond généralement à une somme pondérée des scores obtenus à chacune des questions, est utilisée comme mesure du niveau d'habileté. Ce format de test que nous appellerons «test classique» couvre la majorité des épreuves d'évaluations actuelles. Il se déroule généralement selon la méthode «papier crayon» où l'étudiant inscrit ses réponses,

choisies ou construites, sur une feuille de papier à l'aide d'un crayon ou d'un stylo. Mais, ce format d'examen a beaucoup de limites et pose divers problèmes de précision quant à l'estimation du niveau d'habileté (Wainer, 2000). Parmi ces problèmes, il y a:

- **L'inadéquation de certains items du test à certains apprenants:** en effet, le niveau de difficulté des items auxquels doivent répondre certains apprenants ne correspond pas au niveau d'habileté de ces derniers. Donc, par rapport à son niveau d'habileté dans le domaine concerné par l'évaluation, un apprenant peut faire face à certains items trop faciles ou trop difficiles pour lui. Dans le premier cas, aucun défi n'est relevé par ce dernier, lui laissant l'impression de perdre son temps. Cela peut alors se traduire par des réponses erronées de la part de l'étudiant parce que celui-ci ne se concentre pas sur la tâche qui lui semble sans intérêt. Dans le second cas, lorsque les items sont trop difficiles, l'étudiant peut se décourager au point de ne pas compléter le test. Somme toute, que les items soient trop faciles ou qu'ils soient trop difficiles, un manque de motivation de la part de l'étudiant peut alors se produire avec un impact potentiel sur la précision de l'estimation du niveau d'habileté obtenu.
- **Le manque de précision surtout dans les points extrêmes de l'échelle d'habileté:** cette situation résulte du fait que les tests classiques sont administrés à des apprenants dont le niveau d'habileté varie généralement beaucoup. Donc, ces tests sont constitués d'items dont le niveau de difficulté est assez varié. Pour cette raison surtout, les tests classiques ne permettent pas d'obtenir une estimation précise du niveau d'habileté dans les points extrêmes de l'échelle d'habileté où les valeurs sont très faibles ou très élevées. Ceci est prévisible, car ce type de test est destiné à une large étendue de niveaux d'habileté et se retrouve souvent plus adapté pour les niveaux d'habileté moyens.

- **La facilité de faire du plagiat:** en effet, étant donné que les apprenants reçoivent tous le même test avec la même séquence de questions, il est plus facile de faire du plagiat au moment de l'administration du test.
- **L'absence de mécanismes de détection de patrons de réponse aberrants:** il arrive que des apprenants tentent de masquer volontairement leur vrai niveau dans des évaluations de classement pour différentes raisons (par exemple, se faire classer à un bas niveau en vue de réussir facilement un cours). De plus, la fonctionnalité d'analyse du patron de réponse est importante pour les évaluations diagnostiques afin de s'assurer que les réponses fournies reflètent le vrai niveau d'habileté de l'apprenant.

Comme solution à ces problèmes inhérents aux tests classiques, l'adaptation des tests a été proposée. Cette adaptation des évaluations est un sujet de recherche assez vaste dans le domaine des sciences de l'éducation. Dans la section suivante, quelques approches proposées en éducation pour l'adaptation des évaluations selon le niveau d'habileté cognitive des apprenants seront présentées.

2.2 Propositions d'adaptation des évaluations en éducation

Selon (Wainer, 2000) un test adaptatif consiste à présenter à chaque répondant des questions sur mesure qui sont tributaires de sa maîtrise du sujet, de son profil d'apprenant et de ses réponses aux questions préalablement administrées. Il s'agit donc d'une forme de test sur mesure spécifiquement adapté à la personne à qui on l'administre. Dans un test adaptatif, la première question proposée, les questions subséquentes, l'ordre des questions ainsi que la fin du test peuvent varier d'un apprenant à un autre. Dans les « tests classiques », l'importance est généralement

accordée à la note comme critère pour évaluer le niveau de l'apprenant et faire des inférences par rapport à sa connaissance. Au contraire, dans un test adaptatif c'est la réponse spécifique donnée par l'apprenant à chaque question qui est considérée, ainsi que toute information qui pourra en être déduite ou inférée en cas de réussite ou d'échec à la question. Ceci étant, des questions dont le niveau de difficulté varie peuvent être administrées aux apprenants lors d'une évaluation. Cette caractéristique du test adaptatif fait en sorte que le nombre de bonnes réponses au test ne permet plus de comparer les apprenants entre eux.

Il existe néanmoins un point commun à tous les tests, qu'ils soient classiques ou adaptatifs, c'est qu'ils peuvent être décrits par un ensemble de règles composé de trois éléments. Le premier de ces éléments concerne la façon de déterminer la première question présentée à l'apprenant : ceci constitue *la règle de départ* du test. Le second élément concerne la façon de déterminer quelle sera la question qui suivra une question donnée durant le test : ceci constitue *la règle de suite*. Enfin, le dernier élément consiste à déterminer le moment à partir duquel l'administration des questions doit cesser, en d'autres mots la fin du test : ceci constitue *la règle d'arrêt*. Pour un test classique, ceci pourrait se représenter par le tableau 2.1 ci-dessous :

RÈGLES	ACTIONS
1. Règle de départ	Répondre à une première question, généralement la question #1.
2. Règle de suite	Répondre à une autre question, généralement la suivante.
3. Règle d'arrêt	Terminer le test lorsqu'une réponse a été donnée à la dernière question ou lorsqu'on ne peut plus répondre aux autres questions ou le temps imparti au test est expiré.

Tableau 2.1. Déroulement d'un test classique (Raïche, 2000)

Les trois règles communes aux tests présentées dans le Tableau 2.1 permettront de comparer les différentes propositions d'adaptation des tests.

Dans le domaine des sciences de l'éducation, différentes versions de tests adaptatifs ont été proposées et ont subi de multiples transformations. Dans la prochaine section, un bref survol de certaines des plus importantes versions de tests adaptatifs en éducation sera présenté. Il s'agit du test de Binet, du test à deux étapes et du test pyramidal.

2.2.1 Test de Binet

Généralement, les premiers principes d'application du test adaptatif en éducation sont attribués à Binet et ses collègues au début des années 1900 (Weiss, 1982) et lors du développement des tests d'intelligence chez les enfants. Au départ, l'administrateur détermine, à partir d'informations préalables, l'âge mental du sujet à évaluer. Cette évaluation se fait en utilisant des informations telles que le groupe d'appartenance ou l'âge chronologique. Ensuite, des items dont le niveau de difficulté est approprié à l'âge mental, sont administrés. Si tous les items dont le niveau de difficulté est approprié à l'âge mental, ne sont pas réussis, des items plus faciles appropriés à un âge mental inférieur sont administrés. Sinon, l'inverse est fait. Ceci permet d'obtenir le tableau de règle ci-dessous :

RÈGLES	ACTIONS
1. Règle de départ	Administrer des items dont le niveau de difficulté correspond à l'âge mental de l'enfant
2. Règle de suite	Administrer des items d'un niveau de difficulté supérieur jusqu'à ce que tous les items de ce un niveau aient été échoués. Ensuite, procéder de façon inverse jusqu'à ce que tous les items d'un niveau de difficulté inférieur aient été réussis.
3. Règle d'arrêt	Terminer l'administration du test lorsqu'un âge mental planché et un âge mental plafond ont été déterminés.

Tableau 2.2. Déroulement d'un test de Binet

2.2.2 Test à deux étapes

Un test à deux étapes (*two-stages test*) débute par l'administration d'un test directeur (*routing test*), le même pour tous les répondants. Ceci permettra d'obtenir un premier niveau d'habileté. Par rapport au résultat obtenu dans le test directeur, un second test (*second stage test*), est choisi parmi plusieurs possibilités et est administré afin d'accentuer la précision des résultats obtenus au premier test (de Gruijter, 1980).

RÈGLES	ACTIONS
1. Règle de départ	Administrer le même test à tous les répondants.
2. Règle de suite	Choisir un test de seconde étape selon le résultat obtenu au test directeur.
3. Règle d'arrêt	Terminer le test lorsque le dernier item du second test a reçu une réponse.

Tableau 2.3. Déroulement d'un test à deux étapes

2.2.3. Test pyramidal

Il s'agit d'un test composé d'un ensemble d'items ordonnés en fonction de leur niveau de difficulté dans une structure similaire à une pyramide. Au sommet de la pyramide est placé un item ayant un niveau de difficulté moyen. À l'étage suivant, deux items sont disponibles: l'un des items est de niveau de difficulté légèrement supérieur au précédent, l'autre, d'un niveau de difficulté légèrement inférieur. À chaque étage subséquent, deux items supplémentaires sont ajoutés: un plus difficile et l'autre plus facile. Le test débute par l'administration d'un item de difficulté moyen. Une bonne réponse de l'apprenant entraîne l'administration d'un item de difficulté supérieure. Une réponse erronée, quant à elle, entraîne l'administration d'un item de difficulté inférieure à celle du dernier item proposé. Ceci correspond au tableau des règles 2.4.

RÈGLES	ACTIONS
1. Règle de départ	Administrer le même item de départ de niveau de difficulté moyen à tous les répondants.
2. Règle de suite	Si la réponse est bonne, le prochain item de niveau de difficulté supérieur est administré, sinon le prochain item de niveau de difficulté inférieur.
3. Règle d'arrêt	Terminer lorsqu'un nombre fixe d'items a été présenté.

Tableau 2.4. Déroulement d'un test pyramidal

Une approche de mise en œuvre de test adaptatif par ordinateur basée sur la Théorie de la Réponse à l'Item (Wainer, 2000) est la plus prometteuse. Elle repose sur les propositions modernes et mathématiques de modélisation; elle retiendra notre attention. Après la présentation de la TRI (Théorie de la Réponse à l'Item) au chapitre IV, l'approche de test adaptatif utilisant la TRI implémentée sera expliquée en détail au chapitre VIII.

Malgré ces efforts d'adaptation des évaluations proposés en sciences de l'éducation, dans la majorité des plates-formes actuelles de *E-Learning*, les tests sont encore au format classique. Néanmoins, des efforts commencent à être réalisés en vue de pallier ce problème. Ainsi, la plate-forme de *E-Learning* Moodle offre de faibles capacités d'adaptation des évaluations basées sur une modification du format de présentation des questions avec des pénalités appliquées à chaque tentative de réponse. Mais la séquence des questions du test ne change pas. Des travaux ont également porté sur des systèmes de recommandation de question d'évaluation (Aïmeur et Hage, 2005). Mais l'évaluation en elle-même n'est pas adaptée selon le profil propre à chaque apprenant. Ceci n'est toutefois pas le cas pour l'apprentissage dans lequel des approches intéressantes d'adaptation ont été proposées.

Dans la prochaine section, certaines de ces approches d'adaptation en E-Learning seront présentées. Leurs principes de fonctionnement serviront de d'exemples pour la conception de la plate-forme d'administration d'évaluation adaptative envisagée.

2.3 Approches d'adaptation proposées en informatique

En informatique, parmi les stratégies d'adaptation de l'apprentissage qui sont proposées, on peut citer l'utilisation des systèmes hypermédias adaptatifs, les systèmes d'adaptation selon le profil culturel, etc. À ceux-ci s'ajoutent d'autres paradigmes plus avancés basés sur l'application des techniques d'IA en éducation et sont connues sous l'acronyme AIED (*Artificial Intelligence for Education*). Comme exemple de ces approches, il y a les Systèmes Tutoriels Intelligents (STI) et plus récemment les AIWBES (*Adaptive and Intelligent Web-Based Educational Systems*).

2.3.1 Les systèmes hypermédia adaptatifs

Les travaux d'adaptation via les hypermédias adaptatifs ont été principalement réalisés par Peter Brusilovsky². Les deux principales technologies explorées par cette approche sont la présentation adaptative et le support de navigation adaptatif.

- Le but de la technologie de présentation adaptative est d'adapter le contenu à présenter dans chaque nœud hypermédia (page web) sur trois aspects : l'état des connaissances de l'apprenant, les objectifs visés et toute autre information stockée dans le modèle de l'apprenant. Ainsi, dans un système offrant la présentation adaptative, les pages ne sont pas statiques, mais générées de manières adaptatives et assemblées selon le profil de chaque utilisateur.
- Le but de la technologie adaptative de soutien de navigation est d'aider l'apprenant dans l'orientation et la navigation dans l'hyperespace en changeant l'aspect des liens visibles et accessibles. Par exemple, le système peut de manière

² <http://www.sis.pitt.edu/~peterb/>

adaptative trier, annoter, ou cacher en partie les liens de la page courante pour faciliter ou diriger la navigation vers les pages subséquentes. Le support adaptatif de navigation partage le même objectif que les systèmes d'ordonnancement de programme d'études qui consistent à aider des étudiants à trouver « un chemin optimal » dans le matériel d'apprentissage (par exemple, cours en ligne).

À ces deux principales technologies, s'ajoutent deux autres, notamment :

- Le filtrage adaptatif de l'information (AIF: *Adaptive information filtering*). Il s'agit d'une technologie classique du domaine de la recherche d'informations (*Information Retrieval*). Il vise à trouver les documents qui sont appropriés aux intérêts de l'apprenant dans une grande banque de documents texte. Cette approche est appliquée par les moteurs de recherche pour le filtrage et la recommandation des documents les plus pertinents comme résultats d'une recherche.
- Les systèmes d'apprentissage collaboratif intelligents permettent un support de collaboration en ligne, une création adaptative des groupes d'apprentissage, l'aide par les pairs et une représentation virtuelle des apprenants.

2.3.2 Les systèmes d'adaptation selon le profil culturel

Ces systèmes sont connus sous l'acronyme CAWAS (*Culturally Aware System for eLearning*). D'après Appadurai (1996), l'une des conséquences majeures de la globalisation de l'information est que les individus, partout sur la planète, ont accès au même média. Cependant, le contenu peut être perçu différemment selon la culture locale. Le E-learning, de par son objectif d'être accessible par des apprenants dans un environnement géographiquement distribué, n'échappe pas à cette réalité. Par culture dans l'environnement des CAWAS, on suppose un système relativement stable de concepts et valeurs partagés, un répertoire de symbole significatif, qui procure une expérience, une signification partagée ou une identification avec les autres individus

(Kashima, 2000). Les CAWAS utilisent donc un concept de culture propre à chaque individu qui peut être influencé par différents paramètres appelés les attributs culturels. Comme exemple d'attributs culturels pour une personne donnée, il y a le pays de naissance, le pays de résidence, l'âge, la durée aux endroits visités où il a vécu, l'appartenance religieuse, le groupe ethnique ou racial, la classe sociale, le niveau de scolarité, etc.

Comme principe simplifié de fonctionnement, les CAWAS se basent sur des couples (concept, ressource) à présenter à des apprenants dont certains des attributs culturels précités sont connus. Des algorithmes d'apprentissage machine et des simulations sont utilisés afin d'établir les corrélations entre ces attributs culturels et les affinités des apprenants aux ressources. Les résultats sont utilisés pour faire une adaptation des ressources présentées à l'apprenant dans un cours en ligne en tenant compte de son profil culturel (Resnick et *al.*, 1994).

2.3.3 Les AIWBES

Les AIWBES (*Adaptive and Intelligent Web-Based Educational Systems*) sont des plates-formes web adaptatives qui ont un modèle pour définir les objectifs à atteindre, les préférences et la connaissance de chaque apprenant. Ils utilisent ce modèle dans toute interaction avec l'apprenant. Ceci permet de s'adapter aux besoins particuliers de chaque apprenant. Ces systèmes essaient également d'être plus intelligents en incorporant et en effectuant quelques activités traditionnellement exécutées par un tuteur humain telles que recommander des leçons ou un curriculum particulier à des apprenants ou effectuer des diagnostics (Brusilovsky et Peylo, 2003).

Les AIWBES se trouvent à l'intersection de deux approches d'adaptation utilisées en informatique : les hypermédias adaptatifs et les STI. Les hypermédias adaptatifs offrent la capacité de présenter différentes interfaces, systèmes de navigation,

contenus pour différents apprenants en prenant en considération les informations accumulées sur ces derniers. Par contre, les STI utilisent plutôt des techniques d'IA.

2.4 Conclusion du chapitre

En vue de pallier les problèmes des tests classiques, des efforts ont été fournis en sciences de l'éducation pour proposer des tests adaptatifs. En informatique, même si l'adaptation a fait l'objet de nombreuses recherches et implémentations, les travaux ont principalement été orientés sur l'apprentissage. À l'évidence, l'adaptation devient une préoccupation; différentes propositions et tentatives d'implémentations ont été effectuées. Comme contribution, nous proposons de combiner les acquis en matière d'adaptation de tests en éducation avec des modélisations en informatique. Les STI, qui mettent à contribution les connaissances et interactions entre plusieurs disciplines (les sciences de l'éducation, la psychologie cognitive et l'informatique) constituent un exemple d'une telle approche. L'architecture des STI servira comme modèle dans l'implémentation de PersonFit, raison pour laquelle ce sujet sera présenté dans ce mémoire. Le prochain chapitre présentera les STI ainsi que les aspects de leur architecture qui seront utilisés dans PersonFit.

CHAPITRE III

LES SYSTÈMES TUTORIELS INTELLIGENTS

Ce chapitre nous permettra de présenter les STI dont l'architecture servira de modèle dans le développement de la plate-forme PersonFit. Après la définition et l'explication des principes de fonctionnement, deux modèles implémentés dans les STI seront présentés: le modèle de l'apprenant et le modèle du domaine.

3.1 Définition et fonctionnement

3.1.1 Définition et exemple

Les STI sont des systèmes fournissant un enseignement et un encadrement personnalisés. Ils contiennent des modèles de contenu d'apprentissage qui précisent « *quoi* » enseigner en utilisant des bases de connaissances ainsi des stratégies d'enseignement qui précisent « *comment* » l'enseigner (Wenger, 1987). Ils effectuent des inférences sur l'objet d'apprentissage et les activités qui doivent être réalisées par un apprenant afin d'adapter dynamiquement le contenu ou le style de formation proposé. Dans ce sens, les STI se distinguent des systèmes classiques d'apprentissage ou d'évaluation par ordinateur dans la mesure où ils connaissent la matière à enseigner (*connaissance du domaine*), comment l'enseigner (*connaissance*

pédagogique) et comment acquérir des informations pertinentes sur l'apprenant (*représentation de l'apprenant*). À travers l'interaction entre tous ces composants, les STI permettent de fournir un enseignement adapté à un apprenant sans intervention humaine. Un STI est capable de :

- *diagnostiquer les connaissances de l'apprenant et ses habiletés* en s'appuyant sur des principes plutôt que sur des réponses préprogrammées;
- *décider de la prochaine tâche à exécuter*;
- *adapter l'enseignement ou l'évaluation aux compétences de l'apprenant*;
- fournir une rétroaction sur mesure;
- *dialoguer efficacement avec l'étudiant*;
- encourager la réflexion de la part de l'étudiant.

Les points en italique retiendront particulièrement notre attention dans la conception et l'implémentation de PersonFit.

Un exemple de STI qui nous a fortement inspiré est AutoTutor³. Il a été développé en mode client ensuite déployé sur le Web par une équipe de recherche interdisciplinaire (psychologie, informatique, linguistique, physique et éducation) de l'université de Memphis. Le groupe de recherche a effectué des analyses poussées des stratégies pédagogiques de tutorat humain-à-humain et du discours conversationnel. Cette recherche a fourni les bases empiriques et théoriques pour développer des STI qui aident les étudiants à apprendre en les engageant dans une conversation en langue naturelle sur plusieurs thèmes. Actuellement, AutoTutor existe sous deux versions. La première visant le domaine de l'informatique est conçue pour aider des étudiants à acquérir les connaissances de base en informatique. La seconde est conçue pour aider des étudiants à apprendre la physique newtonienne. Mais, AutoTutor peut également être adapté pour l'usage dans plusieurs autres domaines.

³ <http://www.autotutor.org>

3.1.2 Principe de fonctionnement des STI

Généralement, un STI fonctionne en sélectionnant et en présentant un problème ou une question à l'apprenant. Il compare la solution souhaitée à celle fournie par l'apprenant et effectue ensuite un diagnostic basé sur les différences constatées entre les deux solutions. Après avoir fourni une rétroaction adéquate à l'apprenant, le système fait une nouvelle évaluation, met à jour les informations concernant l'apprenant et répète ce cycle. L'architecture et les fondements soutenant le développement des STI en font un paradigme de développement qui diffère des systèmes adaptatifs présentés dans la section 2.3. Les STI utilisent des techniques d'IA pour assister l'apprenant dans la résolution d'un problème, pour analyser les réponses et effectuer un diagnostic cognitif de l'apprenant.

3.2 Architecture d'implémentation des STI

Quatre modèles principaux se retrouvent dans l'architecture habituelle d'un STI: le modèle de l'apprenant, le modèle du domaine (ou de l'expert), le modèle pédagogique et le modèle de communication. Chacun de ces modèles intègre une ou plusieurs bases de connaissances matérialisées à travers des approches de représentation des connaissances. Cette architecture des STI servira de base pour élaborer celle de la plate-forme PersonFit. Étant donné que l'objectif principal de PersonFit n'est pas d'enseigner, mais plutôt d'évaluer, elle n'inclura ni modèle pédagogique ni modèle de communication. Les deux modèles restants seront présentés en insistant principalement sur les aspects qui seront les plus pertinents pour nos implémentations.

3.2.1 Le modèle de l'apprenant

Le modèle de l'apprenant se définit comme étant la représentation informatique de l'estimation par le système de l'état et du niveau de connaissance de l'apprenant.

Dans un STI, le modèle de l'apprenant comporte deux principaux aspects (Wenger, 1987). Premièrement, il renferme de l'information générale sur l'apprenant: les exercices qu'il a réalisés, la manière dont il les a réalisés (raisonnement, processus de solution), la performance à ces exercices, les sections du système qu'il a parcourues, le temps passé dans chaque section, etc. Deuxièmement, il renferme une représentation de l'état des connaissances explicites ou qui ont été inférées de l'apprenant. Le contenu de ces composantes du modèle de l'apprenant peut varier selon le niveau de granularité de représentation de l'apprenant dans les STI.

Il existe plusieurs stratégies de représentation du modèle de l'apprenant. Mais, nous opterons pour les modèles du type recouvrement (Carr & Goldstein, 1977); car ils sont les plus couramment utilisés. Dans cette approche, le modèle de l'apprenant est conceptualisé comme un sous-ensemble des connaissances du domaine. La figure 3.1 représente une illustration du modèle recouvrement. Sur la figure a, toute la connaissance de l'apprenant est un sous-ensemble de la connaissance globale ou d'un expert dans ce domaine. Une modification de ce modèle pour tenir des connaissances erronées que peut avoir l'apprenant est représentée à la figure b. La formulation du contenu du sous-ensemble peut se faire en terme du niveau d'acquisition (probabilités, niveaux discrets, etc.) par rapport à chaque connaissance.

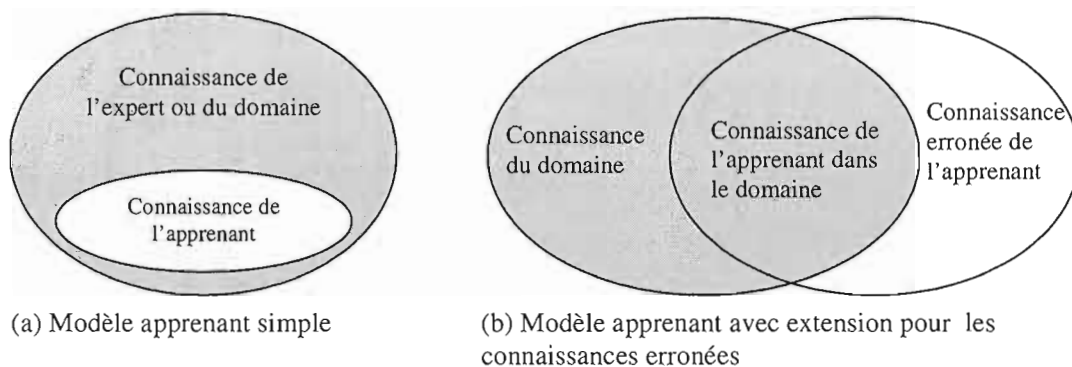


Figure 3.1. Représentation du modèle du type recouvrement, avec une variante incorporant les connaissances erronées de l'apprenant.

3.2.2 Le modèle du domaine

La modélisation des connaissances d'un domaine comprend deux tâches principales. La première est l'analyse du contenu du domaine en termes de concepts, de principes et de procédures (Gagné, 1996). La seconde porte sur l'organisation du contenu formulée en termes de leurs interrelations: par exemple, les relations de pré-requis, de composition, d'analogies, etc. (Nkambou, Frasson et Gauthier, 1996).

3.3 Conclusion du chapitre

Ce chapitre a permis de présenter les STI, de faire un survol de leurs principes de fonctionnement et de présenter les éléments de leur architecture qui seront utiles dans l'implémentation de la plate-forme PersonFit. Les STI utilisent des approches de représentation des connaissances généralement issues de l'IA. Parmi les approches les plus utilisées, nous pouvons citer les représentations purement logiques (Anderson, 1983), les représentations structurelles encore appelées *frames* (Minsky, 1975) et les représentations graphiques. Comme exemple de représentations graphiques, nous avons les réseaux sémantiques (y compris les ontologies), les graphes conceptuels (Sowa, 2000) et les réseaux d'influence (Pearl et Russell, 2003). Étant donné que PersonFit se base sur l'architecture des STI, il nous faudra également rechercher et utiliser des modèles de représentation et de calcul qui permettront de satisfaire les contraintes et objectifs visés par son développement. Dans le chapitre suivant, les modèles choisis seront présentés.

CHAPITRE IV

LES MODÈLES DE CALCUL ET DE REPRÉSENTATION

Dans ce chapitre, les modèles de calcul et de représentation choisis seront présentés. Ils ont été choisis en tenant compte des contraintes de développement notamment:

- La représentation efficace et l'initialisation du modèle de l'apprenant.
- Des métriques et une échelle d'habileté permettant de mesurer et comparer, de façon objective, le niveau d'habileté des apprenants ayant reçu différentes questions lors de l'administration d'un test.
- Des mécanismes permettant de sélectionner les items à administrer à un apprenant pour obtenir le plus précisément possible son niveau d'habileté.
- La possibilité de faire des inférences à partir des informations connues.

En considérant ces contraintes, ainsi que les différents formalismes en IA et les approches d'implémentation de tests adaptatifs utilisés en sciences de l'éducation, deux modèles ont été retenus: les réseaux bayésiens et la théorie de la réponse à l'item. Après la présentation de ces modélisations, une justification de la pertinence de ces choix par rapports aux contraintes précitées sera effectuée. L'interaction entre les deux modélisations afin d'atteindre les objectifs visés sera présentée à la fin.

4.1 La Théorie de la Réponse à l'Item (TRI)

4.1.1 Définition

La TRI est un ensemble de modèles permettant une représentation mathématique probabiliste des caractéristiques des items d'évaluation. Ces caractéristiques sont utilisées pour déterminer le niveau d'habileté d'un répondant à un test (Wainer, 2000).

Le niveau d'habileté désigne le degré d'aptitude ou le niveau de compétence d'un individu dans un domaine précis (par exemple, les connaissances en mathématiques, en français, etc.). Généralement, le niveau d'habileté considéré d'intérêt dans la TRI est d'ordre cognitif et est représenté par la lettre grecque θ (Thêta). Dans la littérature sur la TRI, le niveau d'habileté est également appelé le *paramètre de sujet*.

Il importe de souligner que traditionnellement, dans le domaine de l'évaluation des apprentissages, on utilise plutôt le terme *question*. Le terme *item* quant à lui, désigne beaucoup plus qu'une question. Il pourrait s'agir d'une tâche complexe ou d'une activité auxquelles on pourrait associer une valeur de réussite (1) ou d'échec (0) lors de leurs réalisations. Par la suite dans ce mémoire, le terme item sera utilisé comme faisant référence aux questions.

Comme dispositif central, la TRI fournit des fonctions mathématiques qui relient le niveau d'habileté (θ) et des paramètres associés à l'item à la probabilité d'obtenir un résultat discret (par exemple succès) à cet item. La TRI fournit également des méthodes pour estimer le niveau d'habileté de sujet, les paramètres des items, des techniques pour s'assurer à quel point les données s'adaptent aux modèles, des stratégies pour étudier les propriétés psychométriques des évaluations et la maintenance des banques d'items pour les évaluations. Un avantage de la TRI est

qu'elle fournit potentiellement les informations qui permettent à un chercheur d'améliorer la fiabilité d'une évaluation. Ceci est réalisé par l'extraction d'informations plus sophistiquées concernant les propriétés psychométriques de différents items d'évaluation.

4.1.2 Les fondements de la TRI

Deux considérations servent de fondement pour la TRI. Premièrement, chaque apprenant, dans un domaine dans lequel il subit une évaluation, possède un niveau d'habileté θ donné, auquel on peut associer une valeur sur une échelle d'habileté. En second lieu, par rapport à ce niveau d'habileté θ , cet apprenant a une probabilité $P(u=1|\theta)$ d'obtenir la bonne réponse à un item u selon des caractéristiques propres à cet item. À ces deux considérations s'ajoutent des postulats et hypothèses dépendant du modèle. Ils doivent être vérifiés pour s'assurer de l'adéquation de la modélisation retenue. Au nombre de ceux-ci, nous avons:

- **Une hétérogénéité de la variance** qui signifie que la précision de la valeur estimée du niveau d'habileté lors d'une évaluation n'est pas uniforme dans un intervalle complet d'habileté. Comme conséquence, les scores aux extrêmes de l'échelle du niveau d'habileté ont généralement plus d'erreurs associées que ceux plus près du milieu de l'échelle.
- **Une unidimensionnalité du trait latent** qui est ici le niveau d'habileté. Ceci signifie la réponse obtenue à l'item par l'apprenant est expliquée par un seul niveau d'habileté, d'où l'unidimensionnalité. La multidimensionnalité est toutefois possible dans certaines modélisations récentes de la TRI.
- **Une indépendance locale d'un item à un autre** qui signifie qu'à un niveau d'habileté égal, la probabilité d'obtenir une bonne réponse à un item du test est indépendante de la probabilité d'obtenir une bonne réponse aux autres items.
- **Une invariance du niveau de difficulté des items par rapport aux sujets** qui signifie que la valeur des paramètres associés aux items est indépendante des

sujets qui ont servis à faire le calcul.

- **Une invariance du niveau d'habileté par rapport aux items** qui constitue un grand avantage à utiliser la TRI, car la valeur obtenue du niveau d'habileté est indépendante des items qui ont servi à la calculer.

4.2 La modélisation de la TRI utilisée

Une grande partie de la littérature sur la TRI est consacrée aux différentes modélisations qu'elle offre. Il existe plusieurs modèles dans la TRI. Dans le cadre du développement de la plate-forme PersonFit, c'est le modèle à quatre paramètres noté 4 PL qui sera utilisé. Par la suite, nos affirmations et considérations seront relatives au modèle 4 PL. Ce modèle est assez simple, mais suffisant pour couvrir toutes les implémentations à faire. Il repose sur le postulat de l'unidimensionnalité du niveau d'habileté évalué et des résultats binaires (0 pour une mauvaise réponse et 1 pour une bonne réponse). Il s'agit donc d'un modèle dichotomique et unidimensionnel. Les quatre paramètres du modèle 4PL sont présentés dans la section suivante.

4.2.1 Les paramètres d'items dans le modèle 4PL

Le paramètre de difficulté de l'item, noté b

Ce paramètre affiche presque toujours une corrélation élevée avec la proportion de bonnes réponses observées à un item. Le paramètre b peut théoriquement prendre des valeurs dans l'intervalle de $[-\infty, +\infty]$, mais dans la pratique sa valeur se trouve dans l'intervalle $[-3, 3]$. Il est à noter que la difficulté de l'item et le niveau d'habileté sont mesurés sur le même continuum. Ainsi, il est valide de dire d'un item qu'il est aussi difficile que le niveau d'habileté d'un apprenant et vice-versa. La réussite d'un item de niveau de difficulté b renseigne sur la valeur du niveau d'habileté de l'apprenant.

Le paramètre de discrimination de l'item, noté a

Le paramètre **a** désigne la capacité de l'item à faire la différence entre des personnes dont le niveau d'habileté est inférieur ou supérieur à une valeur seuil θ_s donnée. Le paramètre de discrimination peut théoriquement prendre des valeurs dans $[-\infty, +\infty]$; mais dans la pratique, sa valeur varie généralement entre 0,50 et 2,00.

Lorsqu'un item a un **a** négatif, il devrait être rejeté, car il indique que la probabilité de trouver la bonne réponse à l'item diminue avec le niveau d'habileté.

Le paramètre de pseudo-chance de l'item, noté c

Le paramètre **c** désigne la probabilité pour un répondant dont le niveau d'habileté est faible (près de l'extrémité négatif de l'échelle d'habileté) d'obtenir par chance la bonne réponse à l'item. Étant donné que les items à choix multiples demeurent couramment utilisés dans les tests, le paramètre **c** est important et est employé dans la tentative d'expliquer les effets du hasard sur la probabilité d'une réponse correcte de la part de l'apprenant.

L'asymptote maximale de l'item, noté d

Ce paramètre désigne la valeur vers laquelle tend $P(u=1|\theta)$ pour de grandes valeurs de θ . Si la valeur du paramètre **d** d'un item est inférieure à 1, ceci signifie qu'il n'est pas certain même pour un apprenant très fort, de trouver la bonne réponse à l'item en question. Dans l'implémentation de PersonFit, la valeur du paramètre **d** a été fixée à 1, ce qui est sa valeur par défaut.

4.2.2 La courbe caractéristique d'item

La courbe représentant la probabilité $P(u=1|\theta)$ d'obtenir une bonne réponse à un item **u** en fonction du niveau d'habileté θ est appelée la Courbe Caractéristique de l'Item (CCI). La figure 4.1 ci-dessous est une illustration de la CCI pour deux items.

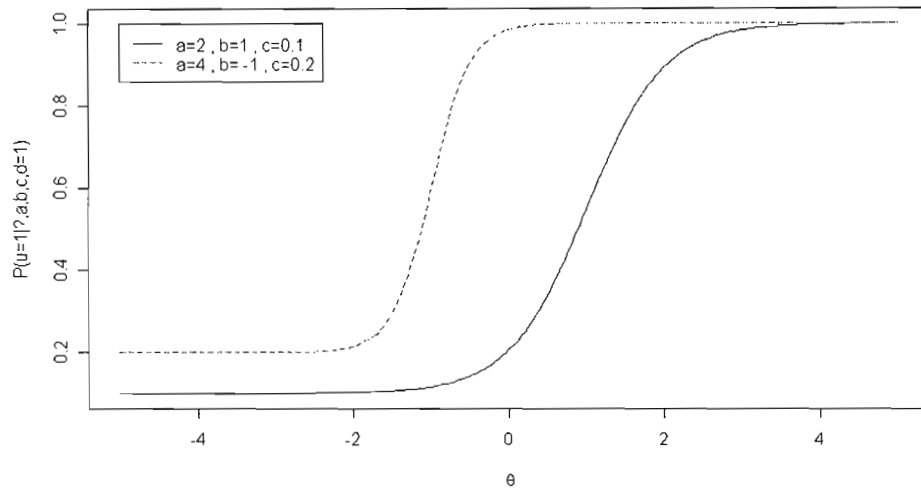


Figure 4.1. CCI de deux items

L'allure des CCI donne des indications sur la valeur des paramètres. Ainsi, la difficulté **b** est indiquée par la valeur de θ qui correspond à $P(u=1|\theta)=0,5$ lorsque le paramètre de pseudo-chance correspond à 0. La pente donne des indications sur la valeur de la discrimination **a**. Plus la valeur de **a** est élevée, plus la courbe présente une pente forte comme le montre la figure 4.1. Théoriquement, un item qui discrimine parfaitement aura donc une pente parallèle à l'axe des ordonnées. L'asymptote inférieure de la CCI correspond à la pseudo-chance **c** et l'asymptote supérieure au paramètre **d**.

4.2.3 Équation du modèle

La formule générale de la probabilité conditionnelle d'obtenir la bonne réponse à un item, connaissant les paramètres d'item (**a**, **b**, **c**, **d**) et le paramètre du sujet (niveau d'habileté θ) selon la modélisation à quatre paramètres est :

$$P(u = 1 | \theta, a, b, c, d) = c + (d - c) \frac{1}{1 + e^{-Da(\theta + b)}}$$

Équation 4.1 Formule du modèle 4PL

$D=1.701$ est une constante qui permet une approximation d'une distribution normale.

$e = 2.71$ est constante mathématiques.

4.3 Stratégies d'estimation des valeurs des paramètres

Différentes stratégies sont utilisées pour calculer la valeur du paramètre du sujet (θ) et celles des paramètres d'item (a , b , c , d).

4.3.1 Estimation du paramètre de sujet (θ)

Si les paramètres des items auxquels un sujet a répondu sont connus de même que son vecteur réponses (*séquence de bonnes et de mauvaises réponses aux items*), l'estimation du niveau d'habileté du sujet peut être effectuée à partir de différentes méthodes mathématiques. Les principales sont : le maximum de vraisemblance (MLE), la méthode bayésienne de maximisation *a posteriori* (MAP) et l'espérance *a posteriori* (EAP). Ces méthodes sont souvent présentées dans des livres sur la TRI dont (Wainer, 2000).

La méthode d'estimation MLE

Son principe consiste à trouver le niveau d'habileté qui maximise la probabilité d'apparition du vecteur réponses. C'est une fonction qui maximise le mode de la distribution d'échantillonnage de l'estimateur du niveau d'habileté qui est ainsi utilisée comme estimateur du niveau d'habileté. Il s'agit donc de maximiser la fonction de probabilité représentée dans l'équation 4.2.

$$P(U \mid \theta) = \prod_{i=1}^n P_i(u_i \mid \theta)^{u_i} Q_i(u_i \mid \theta)^{1-u_i}$$

Équation 4.2 Formule de la méthode d'estimation MLE

Dans cette formule :

n : est le nombre d'items

U : est le vecteur des réponses. Chaque élément $u_i \dots u_n$ a une valeur binaire soit 0 ou 1.

P : est la fonction logistique présentées à l'équation 4.1

$Q = 1 - P$

La méthode MLE a l'avantage d'être assez rapide et de trouver la valeur exacte de θ , mais elle donne de mauvaises valeurs dans plusieurs cas, par exemple:

- si la courbe de distribution des probabilités présente des maxima locaux,
- si le vecteur de réponses est composé uniquement que de 0 ou de 1,
- si on a un petit nombre d'items ($n < 20$),
- lorsque le modèle logistique à trois ou à quatre paramètres est utilisé.

Pour résoudre certains problèmes de MLE, la méthode MAP a été préposée.

La méthode bayésienne de maximisation a posteriori (MAP)

La méthode MAP est un peu moins affectée par les problèmes de maxima locaux. Elle permet le calcul de θ même avec des patrons de réponse où on a uniquement des bonnes réponses ou des mauvaises réponses. Comme principe, on utilise l'information a priori dans le but d'estimer a posteriori le niveau d'habileté, en incorporant cette information a priori (une distribution a priori généralement correspondant à $N(0, 1)$) dans la fonction de maximisation. Sa formule est représentée dans l'équation 4.3

$$P(R | \theta, f(\theta)) = f(\theta) \prod_{i=1}^n P_i(r_i | \theta)^{r_i} Q_i(r_i | \theta)^{1-r_i}$$

Équation 4.3. Formule de la méthode d'estimation MAP

Où $f(\theta)$ est une fonction de probabilité a priori du niveau d'habileté.

La méthode de l'espérance a posteriori (EAP)

La méthode d'estimation EAP se base sur le calcul de l'aire sous la courbe de distribution des probabilités. Son principe consiste à trouver l'espérance mathématique a posteriori du niveau d'habileté. Comme MAP, EAP utilise de l'information provenant de la distribution de probabilité a priori du niveau d'habileté. Plus lente que MLE et MAP, elle offre néanmoins l'avantage de trouver une valeur moyenne de θ près de la bonne valeur. Sa formule générale est donnée par l'équation ci-dessous, en supposant un test de n items.

$$\hat{\theta} = \frac{\int_{-\infty}^{\infty} \theta f(\theta) \prod_{i=1}^n P(u_i = 1 | \theta)^{u_i} P(u_i = 0 | \theta)^{1-u_i} d\theta}{\int_{-\infty}^{\infty} f(\theta) \prod_{i=1}^n P(u_i = 1 | \theta)^{u_i} P(u_i = 0 | \theta)^{1-u_i} d\theta}$$

Équation 4.4. Formule de la méthode d'estimation EAP

Quelle que soit la méthode d'estimation utilisée, l'estimation du niveau d'habileté θ est accompagnée d'une certaine erreur appelée *erreur type* et noté S_{θ} .

Toutes les méthodes d'estimation présentées ont chacune leurs avantages et leurs inconvénients. Les caractéristiques intéressantes pour une méthode d'estimation sont principalement : la facilité de calcul, l'indépendance d'un a priori mal choisi, très efficace pour un bon a priori, non biaisé et peu variable (erreur-type faible). La méthode d'estimation à utiliser dans implémentation de la plate-forme PersonFit sera celle qui nous offrira ces caractéristiques intéressantes précitées. C'est la raison pour laquelle une version adaptative de la méthode MAP (AMAP) a été retenue.

AMAP est une amélioration de la méthode MAP dans laquelle la moyenne de la distribution de la probabilité a priori n'est pas fixée pour toute la durée du test. Cette moyenne au contraire varie et prend la valeur du niveau d'habileté estimé à posteriori après l'administration de l'item $i-1$ (Magis, 2007). Ceci étant, AMAP offre de bons rendements au regard du temps de calcul et de la précision de l'estimateur.

4.3.2 Estimation des paramètres des items

Le processus d'estimation des paramètres des items porte le nom de *calibrage des items*. Deux situations classiques peuvent se présenter: La première, lorsque la valeur de θ est connue et la seconde, qui procède par estimation conjointe du paramètre de sujet θ et des paramètres d'item.

En supposant, dans la première situation que les niveaux d'habileté θ des apprenants sont connus, l'objectif est de trouver les paramètres d'un nouvel item auquel ils ont répondu. Plusieurs méthodes d'estimation sont alors offertes par la TRI. Parmi les plus couramment utilisées, il y a l'estimation par maximum de vraisemblance et l'estimation par maximum de vraisemblance a posteriori.

Le second cas est l'estimation conjointe du niveau d'habileté et des paramètres des items. Ce cas concret se présente lorsqu'on veut transformer une épreuve d'évaluation classique dont les items et les réponses des apprenants à ces items sont connus en évaluation à items modélisés selon la TRI. Comme méthode d'estimation conjointe proposée par la TRI, il y a le JMLE (Join Maximum Likelihood Estimation). Comme principe, le JMLE doit disposer d'un fichier de résultat aux items contenant M répondants pour N items sur lequel on applique un prétraitement. Par exemple, retirer les items qui ont reçu de manière constante soit une bonne, soit une mauvaise réponse de la part de tous les sujets (car il serait difficile de calculer leur niveau de difficulté). À la première étape, les paramètres de sujets sont ramenés à une distribution normale $N(0, 1)$, ensuite les paramètres d'items sont estimés par une des méthodes par maximum de vraisemblance, ce qui permettra en deuxième étape d'estimer les paramètres des sujets. Ce processus est répété de manière itérative jusqu'à atteindre une convergence.

Il existe actuellement des logiciels puissants permettant le calibrage des items selon différents modèles logistiques. Parmi les plus connus et utilisés, il y a *Bilog*⁴. Afin de faciliter l'utilisation de la plate-forme PersonFit et pour éviter les dépendances avec d'autres logiciels, il est prévu d'y intégrer de façon optionnelle des fonctionnalités de calibrage d'items.

Mis à part le niveau d'habileté et les paramètres d'items, la TRI offre d'autres indices statistiques. Certains d'entre eux nous seront utiles dans l'implémentation de PersonFit. Le plus importante est l'information au sens de Fisher et l'indice d'ajustement L_z .

4.4 Autres indices de la TRI à utiliser

4.4.1 L'information

L'information au sens de Fisher est un indicateur de la précision avec laquelle le niveau d'habileté est estimé. Ceci constitue une des contributions principales de la TRI en proposant une extension du concept de la fiabilité. Traditionnellement, la fiabilité se rapporte à la précision de la mesure (c'est-à-dire, le degré avec lequel une mesure est exempte d'erreur). La fiabilité est mesurée en utilisant un index qui peut être défini de diverses manières, mais qui suppose une homogénéité de la variance. Toutefois, selon l'hypothèse de l'hétérogénéité de la variance (section 4.1.2), la TRI indique clairement que la précision n'est pas uniforme à l'intérieur d'un intervalle complet de score à une évaluation. Pour ce faire, la TRI utilise donc le concept d'information d'item et de test pour remplacer le concept de fiabilité.

L'information est également une fonction des paramètres du modèle. Par exemple, selon la théorie de l'information de Fisher, la fonction de l'information d'un item i

⁴ <http://work.psych.uiuc.edu/irt/downloads.asp>

pour le modèle à trois paramètres (**a**, **b**, **c**) est donnée par la formule de l'équation 4.6. L'information au sens de Fisher n'est calculée qu'avec les estimations axées sur la vraisemblance maximale, par exemple à l'intérieur de la MLE et de la MAP. Dans l'équation 4.6, nous pouvons remarquer l'effet du paramètre de discrimination. Les items fortement discriminatifs ont des fonctions d'information grandes et étroites; ils contribuent considérablement pour le calcul du niveau d'habileté de l'apprenant, mais sur une gamme étroite de l'échelle d'habileté. Au contraire, un item moins discriminatif fournit moins d'informations, mais sur une plus grande gamme de l'échelle d'habileté.

$$I_i(\theta) = a^2 \left[\frac{Q_i(\theta)}{P_i(\theta)} \right] \left[\frac{P_i(\theta) - c^2}{(1 - c^2)} \right]$$

Équation 4.6. Fonction d'information pour un item dans le modèle 3 PL

4.4.2 L'indice d'ajustement Lz

Il existe, dans la TRI, des méthodes pour déterminer des vecteurs réponses anormaux par rapport aux items présentés. Ces méthodes calculent un indice d'ajustement (*fit*) du vecteur réponse de l'apprenant au modèle. Au nombre de ces méthodes, il y a l'indice Lz. Pour un test constitué de n items, lorsque la probabilité d'obtenir une bonne réponse à chacun des items $P(u = 1|\theta)$ est obtenue selon la modélisation 4PL, la formule de l'indice Lz est donnée par l'équation 4.7

$$l_z = \frac{l_0 - E(l_0)}{[Var(l_0)]^{1/2}}$$

Équation 4.7 Formule de l'indice Lz

Dans cette équation,

L_0 représente le logarithme de la vraisemblance du patron de réponse. Sa formule est donnée par l'équation 7.2.

$$l_0 = \sum_{i=1}^n \left[u_i \ln P(u_i = 1 | \theta) + (1 - u_i) \ln P(u_i = 0 | \theta) \right]$$

Équation 4.8 Formule de L_0

$E(L_0)$ représente l'espérance mathématique. Sa formule est donnée par l'équation 7.2.

$$E(l_0) = \sum_{i=1}^n \left[P(u_i = 1 | \theta) \ln P(u_i = 1 | \theta) + P(u_i = 0 | \theta) \ln P(u_i = 0 | \theta) \right]$$

Équation 4.9 Formule de $E(L_0)$

$Var(L_0)$ représente la variance. Sa formule est donnée par l'équation 7.3.

$$Var(l_0) = \sum_{i=1}^n \left\{ P(u_i = 1 | \theta) P(u_i = 0 | \theta) \left[\ln \frac{P(u_i = 1 | \theta)}{P(u_i = 0 | \theta)} \right] \right\}$$

Équation 4.10 Formule de $Var(L_0)$

L'indice L_z est interprété comme un score Z , les valeurs extrêmes se situent en dessous du seuil fixé à -1,96. Donc, généralement un patron de réponses est jugé inadéquat par rapport aux items proposés lorsque sa valeur de L_z est fortement négative ($< -1,96$).

4.5 Conclusion sur la TRI

Somme toute, la TRI est une modélisation qui présente de nombreuses caractéristiques qui répondent aux exigences d'implémentation de PersonFit. Par exemple, la TRI fournit des méthodes pour estimer efficacement le niveau des habiletés cognitives des apprenants. Ces valeurs sont obtenues sur une échelle d'habileté et permettent de comparer de façon objective des apprenants ayant reçu différentes questions lors d'un test. La TRI fournit également des stratégies pour estimer les paramètres des items à partir des données. Elle offre différents indices statistiques, dont l'information qui pourra servir comme critère de sélection d'un item optimal à administrer. D'autres outils offerts par la TRI permettent d'établir à quel point les données s'adaptent à un modèle et permettent d'étudier les propriétés psychométriques des évaluations.

Malgré les avantages précités de la TRI, les informations qu'elle fournit ont une portée locale. Par exemple, le niveau d'habileté estimé (θ), son erreur-type (S_θ) et l'indice d'ajustement du patron de réponse (L_z) sont relatifs au test actuel et aux concepts sur lesquels porte ce dernier. Les modèles unidimensionnels de la TRI n'offrent pas de mécanisme permettant de faire des inférences sur les habiletés potentielles de l'apprenant dans d'autres concepts connexes à ceux pour lesquels son niveau d'habileté est connu. Hors, ceci est important pour l'initialisation du modèle de l'apprenant au début d'un autre test. Également, selon les théories des sciences cognitives, un apprenant a besoin de connaître non seulement des concepts spécifiques, mais également des relations comme la similarité, la différence, l'agrégation, etc. entre les concepts. Il serait difficile d'évaluer un apprenant sur tous ces aspects. Pour les raisons précitées, un autre modèle doit donc être utilisé comme complément de la TRI. Les réseaux bayésiens avec leur capacité d'inférence viennent donc comme une modélisation complémentaire à la TRI. Cette modélisation sera présentée dans la prochaine section.

4.6 Les réseaux bayésiens (RB)

4.6.1 Définition

Les RB sont des graphes orientés et acycliques dont les nœuds sont des variables aléatoires et dont la structure traduit les dépendances conditionnelles entre ces variables. Les RB sont à la fois des modèles de représentation des connaissances et une "machine à calculer" les probabilités conditionnelles.

Pour un domaine donné (par exemple médical), les RB permettent de décrire par un graphe les relations causales entre variables d'intérêt. Dans ce graphe, les relations de cause à effet entre les variables ne sont pas déterministes, mais probabilistes. Ainsi, l'observation d'une ou de plusieurs causes n'entraîne pas systématiquement les effets qui en dépendent, mais modifie seulement la probabilité de les observer. De plus, on dispose d'algorithmes pour propager dans le reste du réseau l'information ou les évidences à partir d'un nœud dans le reste du réseau. L'intérêt particulier des RB est de tenir compte simultanément de connaissances *a priori* d'experts (dans le graphe) et de l'expérience contenue dans les données. Les RB sont utilisés par exemple pour le diagnostic (médical ou industriel) et l'analyse du risque. En les utilisant, on peut, en fonction des manifestations observées, calculer la probabilité des données non observées. En guise d'illustration, un exemple simple de RB est présenté dans la prochaine section.

4.6.2 Exemple simple de réseau bayésien

La figure 4.2 ci-dessous est une illustration d'un RB représentant les dépendances entre cinq variables. Les cinq variables en question sont des opérations arithmétiques de base à savoir : l'addition, la soustraction, la multiplication, la division et la simplification (de fraction). Ils sont représentés respectivement par X_1 , X_2 , X_3 , X_4 et X_5 . Toutes les autres variables ont des valeurs binaires : connaissance ou pas

connaissance de l'opération concernée. En suivant les effets indiqués par les arcs, la connaissance de la soustraction et de la multiplication a un impact sur la division. Les relations entre ces différentes variables peuvent être probabilisés.

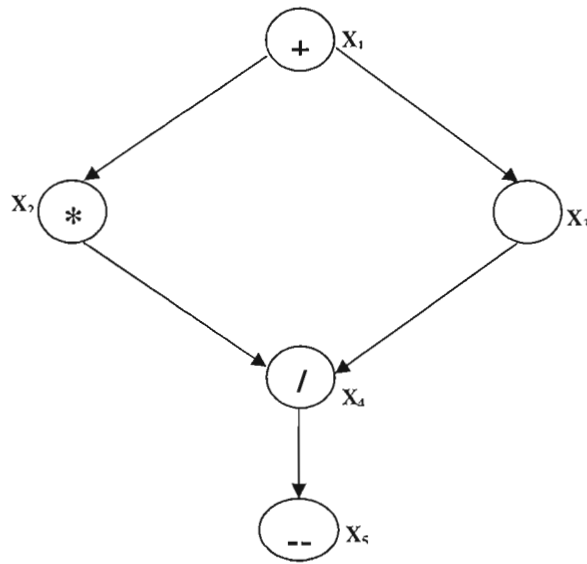


Figure 4.2. Un RB représentant les dépendances entre cinq variables

L'équation 4.11 représente la probabilité conditionnelle du réseau de la figure 4.2.

$$P(X_1, \dots, X_5) = P(X_1) * P(X_2|X_1) * P(X_3|X_1) * P(X_4|X_2, X_3) * P(X_5|X_4)$$

Équation 4.11. Probabilité conditionnelle entre cinq variables

Lorsqu'une variable (par exemple X_2) exerce une causalité sur une variable (par exemple X_4), alors on peut exprimer la probabilité conditionnelle d'obtenir X_4 en connaissant X_2 . Il s'agit dans ce cas d'une *déduction bayésienne*. Au contraire, si X_4 est observé, alors la probabilité de X_2 sachant X_4 est donnée par des formules de vraisemblance. Il s'agit dans ce cas d'une *abduction bayésienne*.

En prenant en considération la possibilité de représentation probabiliste, d'inférence et d'abduction offerte par les RB, ils apparaissent pertinents pour représenter les

informations probabilistes obtenues sur les apprenants et faire des inférences à partir de celles-ci. Dans la section suivante, l'utilisation des RB comme modélisation dans PersonFit sera présentée.

4.6.3 Utilisation des RB dans PersonFit

En règle générale, dans le domaine des STI, les RB ont tendance à être de plus en plus utilisés pour la gestion de l'incertitude en modélisation de l'apprenant. Ceci s'explique par leur sémantique relativement claire et les possibilités d'apprentissage machine qu'ils offrent. Dans PersonFit, le modèle de l'apprenant représente les connaissances relatives à ce dernier. Un certain nombre de ces connaissances sont explicites, provenant d'une prise en compte directe des résultats de l'apprenant après un test. D'autre part, il y a les connaissances qui sont implicites et sont inférées par des mécanismes propres au modèle (Myslevy et Gitomer, 1996). Ces inférences sont par nature incertaines, tout comme le sont les déductions que peut faire un tuteur à partir des résultats aux tests de ses apprenants. Par exemple, une même erreur peut provenir de différentes causes sans qu'il soit nécessairement possible de déterminer laquelle s'est appliquée.

Pour l'implémentation des RB dans PersonFit, les nœuds sont les concepts d'un domaine dans lequel l'apprenant a subi des évaluations. La valeur du niveau d'habileté de l'apprenant dans ces concepts est utilisée pour déterminer la probabilité d'obtenir son patron de réponses. Cette probabilité est alors associée au nœud correspondant dans le RB. Étant donné que le modèle 4PL utilisé suppose une unidimensionnalité du niveau d'habileté mesuré, les tests doivent porter sur des concepts relatifs à un seul domaine (par exemple informatique, français, mais pas les deux en même temps). Il n'y donc pas de risque d'associer la même valeur du niveau d'habileté obtenue après un test à des concepts de domaines différents. Par la suite,

on pourrait bénéficier des différentes possibilités d'inférences offertes par le RB pour estimer le niveau d'habileté dans d'autres concepts du même domaine.

4.7 La complémentarité entre TRI et les RB dans PersonFit

Dans PersonFit, la TRI et les RB seront utilisés de façon complémentaire. La TRI et les RB sont deux modèles probabilistes, ce qui est convenable pour la nature parfois incertaine des données et des informations manipulées. La raison principale de notre choix des RB provient de son utilité pour la modélisation de l'apprenant. Les RB ont été proposés dans plusieurs articles comme solution pour gérer l'incertitude dans la modélisation de l'apprenant dans les STI (Conati *et al.*, 2002).

En ce qui concerne la TRI, elle offre également de nombreux avantages. Selon (Woolf *et al.*, 2005), les stratégies utilisées pour l'évaluation des habiletés cognitives de l'apprenant peuvent se diviser en deux grandes catégories : l'approche basée sur l'expert et l'approche orientée données. La première, soit l'approche experte se base sur un modèle de l'expert pour identifier les compétences nécessaires pour résoudre un problème. On se base alors sur les connaissances d'un expert pour déterminer la structure, le modèle et si possible, les paramètres des items d'évaluations proposés à l'apprenant. La seconde catégorie, soit l'approche orientée données, se base au contraire sur les données ou informations disponibles sur les résultats des apprenants pour faire ressortir des structures et des équations permettant de relier l'habileté d'un apprenant à sa performance. La TRI est un exemple de cette catégorie. Des études (Woolf *et al.*, 2005) ont démontré les avantages de la TRI en matière de complexité d'implémentation, de pouvoir prédictif et d'indépendance d'un jugement subjectif voire biaisé, d'un expert. Un autre grand avantage offert par la TRI est que l'estimateur du niveau d'habileté obtenu est indépendant des items qui ont servi à le calculer.

4.8 Des approches intéressantes non utilisées

Il existe certaines propositions émergentes d'implémentation de test adaptatif basées uniquement sur une architecture de RB sans couche cachée nommée POKS (Desmarais et Pu, 2005). Des études comparatives (Desmarais et Pu, 2005) montrent que POKS fournit des performances semblables au modèle à deux paramètres (*difficulté* et *discrimination*) de la TRI pour la classification des apprenants. Mais POKS ne fournit pas une échelle de mesure, ce qui est important en évaluation des apprentissages. De plus, POKS est un modèle plus récent que la TRI et n'a pas encore fait ses preuves dans des implémentations réelles. Pour ses raisons, POKS n'a pas été retenue comme modélisation. Une combinaison TRI et RB s'est avérée une meilleure alternative pour nous car ces deux modèles sont matures et ont fait leurs preuves dans des implémentations réelles dans leur domaine respectif, soit la psychométrie et l'informatique.

4.9 Conclusion du chapitre

Dans ce chapitre, les modèles de calcul et de représentation retenus pour l'implémentation de PersonFit ont été présentés. Il s'agit de la TRI et des RB. Une collaboration entre ces deux modélisations permet de répondre aux contraintes imposées pour le développement de PersonFit. Après avoir décrit ces deux modélisations qui répondent aux exigences de conception de la plate-forme, la prochaine étape va consister à définir les principes qui doivent diriger son implémentation, de même que les fonctionnalités à offrir. Ces deux aspects seront abordés en détail dans le prochain chapitre.

CHAPITRE V

PRINCIPES DIRECTEURS ET FONCTIONNALITÉS DE PERSONFIT

Ce chapitre marque le début des chapitres techniques portant sur l'implémentation de PersonFit. Il va permettre de présenter les principes qui ont servi de fondements pour l'implémentation de la plate-forme ainsi que les fonctionnalités prévues.

5.1 Objectifs et fonctionnalités

L'objectif de conception de PersonFit est basé sur des préoccupations réelles. Entre autre, il y a la volonté d'étendre les fonctionnalités de *E-testing* des plates-formes actuelles de *E-learning* à l'administration de tests adaptatifs. Ainsi, sans être une solution clé en main ou exhaustive, nous visons tout au moins à fournir des solutions pratiques et assez simples d'utilisation. Il existe néanmoins des contraintes connexes à considérer dans ce développement. Il s'agit notamment de pouvoir gérer une grande banque d'items qui serviront aux évaluations. Cette banque doit pouvoir être alimentée en items par les participants au projet. Un moteur intelligent d'administration de test pourra se connecter à la banque et sélectionner les items les plus adaptés pour évaluer chaque apprenant. Ces items doivent donc être

indépendants d'une plate-forme spécifique, doivent avoir des métadonnées permettant leurs manipulations et affichages de façon adaptative.

PersonFit devrait donc permettre aux professeurs participants de créer des items d'évaluation sous un format normalisé pour alimenter la banque partagée d'items. Généralement, les professeurs disposent localement d'une banque d'items d'évaluation au format Microsoft Word ou un autre éditeur de texte. Dans ce cas, des fonctionnalités pour l'importation de ces items du format Microsoft Word vers le format adopté seront offertes.

Somme toute, sur la base de ces objectifs et des autres contraintes d'implémentation, la plate-forme PersonFit offrira les fonctionnalités suivantes :

- La création d'items d'évaluation sous un format respectant les standards actuels dans le domaine du *E-learning*.
- L'importation des items conçus en utilisant Microsoft Word ou d'autres logiciels populaires d'édition de questionnaires comme NetQuiz⁵ ou Hot Potatoes⁶ dans le format adopté.
- Le chargement et le téléchargement de ces items dans la plate forme Moodle.
- La génération de tests adaptatifs selon différents niveaux d'habileté.
- La visualisation et simulation de ces tests hors ligne par le professeur.
- La simulation des résultats à un test adaptatif sur la base d'exemplaire de modèles d'apprenants qui peuvent être récupérés dans Moodle.
- Des manipulations relatives à la TRI notamment :
 - La simulation de patrons de réponses selon différents modèles logistiques.
 - La détection de patrons de réponses inappropriés.

⁵ <http://www.ccdmd.qc.ca/ri/netquizpro/>

⁶ <http://hotpot.uvic.ca/>

- La visualisation des courbes caractéristiques des items ainsi que de leurs fonctions d'information.
- La création de différents fichiers de données et de traitements.

Dans son implémentation, la plateforme PersonFit est constituée de deux systèmes distincts, mais dépendants:

1. Un logiciel utilitaire, développé en Java et fonctionnant en mode client. Ce logiciel est utilisé par les professeurs ou toutes personnes autorisées à alimenter la banque d'items. Parmi les fonctionnalités offertes par cet utilitaire, il y a la création d'items. Également, une taxonomie des objectifs d'apprentissages propres à chaque domaine peut être élaborée afin de joindre à chaque item les objectifs d'apprentissage pertinents. Par défaut, la taxonomie de Bloom (Bloom *et al.*, 1956) est présentée. Ce logiciel permet également la gestion de la banque locale d'items, le chargement et le téléchargement de cette banque vers Moodle, la création des tests adaptatifs et la simulation de ces tests sur la base du modèle des apprenants qui peuvent être récupérés dans Moodle.
2. Le second système est constitué des modules développés et modifiés de la plate-forme Moodle pour la gestion de la banque partagée d'item. Ces items sont créés en utilisant le logiciel utilitaire client ou l'interface de création de questionnaire intégrée de Moodle. Certaines fonctionnalités du logiciel client ont été intégrées dans Moodle permettant l'administration en ligne de tests adaptatifs aux apprenants.

Le nom donné à la plate-forme, *PersonFit*, résume les objectifs et les fonctionnalités proposées. L'expression '*Person*' est utilisée pour désigner une personne mais plus précisément un apprenant. L'expression '*fit*' est un terme anglais dont la traduction en français la plus adéquate dans notre contexte est 'ajustement'. Plus spécifiquement

dans notre cas, il s'agit de faire un ajustement et donc une adaptation des évaluations selon le niveau d'habileté et le profil des apprenants. Également, il faudrait étudier l'ajustement de son patron de réponse pour faire une détection possible de jeu.

Pour mener à bien l'implémentation de PersonFit en y intégrant toutes les fonctionnalités précédemment décrites, des principes directeurs ont été fixés et doivent guider nos choix technologiques. Ces principes seront présentés dans la section suivante.

5.2 Les principes directeurs

Les choix technologiques pour l'implémentation ont été effectués sur la base de certains principes. Ces principes s'inscrivent non seulement dans notre vision de l'informatique, mais également conviennent pour faciliter l'évolution du projet, les éventuelles futures collaborations ainsi que la mise à disposition de la plate-forme pour d'autres utilisateurs sans contraintes de droits d'auteur.

Deux principes ont été fondamentaux retenus. Le premier est la conformité aux normes actuelles dans le domaine de E-learning. Le second consiste en l'intégration des fonctionnalités développées dans une plate-forme de E-learning existante et populaire afin de vulgariser les travaux effectués. Les deux prochaines sous-sections portent sur les normes de E-learning et sur la plate-forme d'intégration utilisées.

5.2.1 Les normes du E-learning utilisées dans PersonFit

Les normes sont les seuls moyens qui permettent l'interopérabilité et l'évolution des systèmes au cours du temps. Ainsi, l'indexation et l'annotation des items d'évaluation seront réalisées sur la base des normes actuelles dans le domaine du E-learning. Ceci va permettre de garantir les échanges, l'interopérabilité, la réutilisation et

l'adaptation. Les normes Dublin Core et LOM et les spécifications de IMS ont été principalement utilisées.

Les méta-données Dublin Core

Le Dublin Core⁷ (ISO 15836:2003) est une norme internationale qui fournit un ensemble simple et normalisé de conventions pour décrire des ressources en ligne de manières qui facilitent leur recherche, utilisation et partage. Il est employé couramment pour décrire les fichiers numériques tels que la vidéo, l'audio, l'image, le texte et les médias composés comme des pages Web. Son implémentation se fait généralement sous forme XML. L'expression « Dublin » dans son nom se rapporte à la ville de Dublin, Ohio aux États-Unis, où les premiers travaux sur cette norme ont débuté en 1995. La sémantique du noyau de Dublin a été établie et est maintenue par un groupe international de professionnels multidisciplinaires. Le noyau de la norme Dublin inclut deux niveaux: le noyau simple et le noyau qualifié. Le noyau simple est suffisant pour nos besoins d'implémentation. Il comporte quinze éléments. Le tableau 4.1 montre certains de ces éléments, regroupés par catégorie d'informations décrites. Les éléments des catégories *Contenu* et *Propriétés intellectuelles* feront partie des métadonnées qui vont être utilisés pour la description des items dans PersonFit.

Nom de l'élément en français	Catégorie
Titre, Sujet, Description, Source, Langue	Contenu
Créateur, Éditeur, Contributeurs, Droits	Propriétés intellectuelles
Date, Type, Format, Identifiant	Instance particulière

Tableau 5.1. Quelques éléments du noyau simple de Dublin

⁷ <http://dublincore.org/>

Une extension du Dublin Core a été effectuée et normalisée IEEE-1484.12.1-2002. Elle est principalement destinée à la description des Objets d'Apprentissages (OA). Il s'agit de la norme LOM.

La norme LOM (Learning Object Metadata)

La norme LOM est structurée de manière hiérarchique à l'aide de 78 éléments en trois niveaux et neuf catégories. Les catégories LOM qui seront utilisées sont:

- **Général** : qui permet la description et l'identification de la ressource.
- **Pédagogique** : qui accomplit une fonction de description pédagogique de la ressource. On y retrouve par exemple le contexte d'utilisation (scolaire, secondaire, supérieur, formation continue, etc.), le type de public cible (enseignant, auteur, apprenant, gestionnaire), la difficulté, l'âge des apprenants auxquels la ressource est destinée, etc.
- **Droits** : Il s'agit des conditions légales d'utilisation de la ressource: indications de coût, de licences, de droits d'auteur.
- **Classification** : qui permet d'assigner à une ressource des classifications et des vocabulaires contrôlés. Des classifications peuvent être attribuées pour la discipline, les objectifs pédagogiques, les restrictions, les compétences.

Certains éléments des catégories droits seront utilisés pour la gestion des droits d'utilisation et de partage des items lors de leur conception dans PersonFit. Également, un vocabulaire contrôlé basé sur la catégorie classification sera utilisé pour les mots clés associés au item de PersonFit.

Les spécifications de IMS (Instructional Management Systems)

IMS Global Learning Consortium est un groupe de travail né en 1997 qui est composé de membres issus de différents secteurs d'activités (éducation, entreprises et organisations gouvernementales). L'IMS a principalement deux objectifs. Le premier est de définir des spécifications techniques pour l'interopérabilité des applications et services en E-learning. Le second est de supporter l'incorporation de ces spécifications dans les technologies du Web utilisées en E-learning. L'une des spécifications de à savoir le IMS-QTI (*Question & Test Interoperability*) définit des formats XML et méthodes de traitement pour les items et les évaluations. Dans PersonFit, la version 2 de IMS-QTI sera utilisée comme format XML pour le stockage physique des items.

Le Dublin Core, le IEEE-LOM et le IMS sont assez stables et couramment utilisés dans l'industrie du *E-learning*. Ils présentent toutefois certains problèmes provenant de leurs principes de fonctionnement basés sur les métadonnées. La prochaine section portera sur un de ces problèmes auquel nous aurons à faire face dans l'implémentation de PersonFit.

5.2.2 Problèmes de confiance dans les métadonnées des normes actuelles

À l'exception des données physiques (la date de création, le type, le format, etc.), la majorité des métadonnées des normes qui seront utilisées peuvent être qualifiées de subjectives. En effet, les métadonnées telles que le sujet, la description, la relation, etc. sont issues de l'appréciation subjective du concepteur. Ceci est directement à l'origine d'un problème, celui de la confiance, c'est-à-dire la confiance que les utilisateurs (personnes, organismes ou systèmes) peuvent accorder aux créateurs des métadonnées. Dans le cadre des normes mentionnées, toute la responsabilité revient aux créateurs des métadonnées de fournir une description aussi précise que possible.

Mais l'expérience dans d'autres domaines presque similaires, démontre qu'il y a une faiblesse problématique. L'exemple concret est celui du standard HTML qui permet aux concepteurs d'inclure dans leur document des métadonnées qui décrivent le contenu de la page. Ceci se fait à travers les balises « meta » dans l'entête du document. Le but de ces balises « meta » dans des documents HTML était (et reste) identique au but des métadonnées dans les normes actuelles à savoir l'indexation, les recherches, etc. Mais leur utilisation a été un échec et plusieurs moteurs de recherches, qui utilisent uniquement ces informations pour la recherche, obtiennent de piètres résultats. Cet état des choses s'explique par le fait que ces métadonnées sont volontairement ou involontairement mal fournies, ou bien sont plus promotionnelles que descriptives.

Dans PersonFit, pour régler ce problème, les concepteurs d'items seront regroupés en communauté de confiance (*trust community*) ayant un modérateur comme dans les forums de discussions. Une communauté de confiance peut, par exemple regrouper des enseignants d'un même cours, qui se connaissent ou se font mutuellement confiance pour les partages d'items. Ils peuvent décider d'alimenter une banque commune d'items d'évaluation. Au-delà d'accorder les privilèges des membres et les droits de partage associés aux items, le modérateur doit être désigné pour son habileté à valider le contenu et les métadonnées de tous les items avant leur ajout à la banque d'items.

Somme toute, des mesures ont été prises comme solutions aux problèmes de confiance relatifs aux métadonnées des normes qui seront utilisées. L'intégration de ces normes va permettre l'interopérabilité des données de PersonFit, son évolution et sa vulgarisation. C'est d'ailleurs dans cette optique de vulgarisation que le second principe directeur d'implémentation de PersonFit, à savoir l'intégration à une plateforme existante et libre, entre en jeu. La plateforme retenue sera présentée dans la prochaine section ainsi que les raisons de ce choix.

5.2.3 La plate-forme libre retenue pour la vulgarisation de PersonFit

Comme stratégie pour rendre PersonFit accessible à un grand nombre d'utilisateurs, nous avons décidé d'intégrer une partie de ces modules dans une plate-forme de E-learning libre et populaire. La plate-forme Moodle⁸ a été retenue.

Moodle est une plate-forme de *E-learning* sous licence GPL servant à créer des communautés d'apprenants autour de contenus et d'activités pédagogiques. C'est un projet bénéficiant d'un développement actif et conçu pour favoriser un cadre de formation constructiviste. Le terme « Moodle » est un acronyme pour «*Modular Object-Oriented Dynamic Learning Environment*». Moodle fut créé par Martin Dougiamas⁹ dans le cadre de ses recherches doctorales à l'Université de Curtin en Australie. Il présenta sa première version au mois d'août 2002. La plate-forme en est actuellement à sa version 1.8.

Moodle présente de nombreux avantages qui ont motivé son choix comme plate-forme expérimentale. Au nombre de ceux-ci, nous avons:

- La popularité qui provient de sa stabilité et d'une grande communauté qui contribue à son développement, sa maintenance et son évolution. Un nombre grandissant d'institutions d'enseignement l'adopte comme plate-forme de E-learning. L'UQAM (*Université du Québec À Montréal*) en offre un exemple, ayant récemment migré de WebCT¹⁰ à Moodle comme plate-forme de E-learning.
- De nombreuses fonctionnalités fort intéressantes comme les forums de discussion, les gestionnaires de ressources, les interfaces graphiques et conviviales pour concevoir des questionnaires, des tests, des sondages.

⁸ <http://moodle.org/>

⁹ <http://dougiamas.com/>

¹⁰ <http://www.webct.com/>

- Une grande modularité. Comme son acronyme l'indique, ceci constitue le fondement de sa conception. Il est donc facile de développer ses propres modules à ajouter à Moodle ou de modifier les modules déjà existants.
- Moodle a été développé avec PHP qui est un langage gratuit. Également, il peut fonctionner avec la majorité des bases de données libres (en particulier MySQL).
- Même si l'anglais est la langue de base de développement de Moodle, ce dernier comporte une traduction de l'interface usager dans plusieurs langues dont le français canadien (fr_CA).

5.3 Conclusion du chapitre

Ce chapitre a permis de présenter les principes qui vont servir de fondements pour l'implémentation de la plate-forme PersonFit ainsi que les fonctionnalités qui y sont prévues. Le prochain chapitre décrira l'utilisation de ces principes ainsi que des modélisations retenues en vue du développement de la plate-forme.

CHAPITRE VI

LES COMPOSANTS DE PERSONFIT ET LEURS INTERACTIONS

Ce chapitre servira à une description détaillée du fonctionnement de PersonFit ainsi qu'à présenter l'intégration des différentes modélisations et normes utilisées. L'algorithme de déroulement de test adaptatif utilisé sera présenté et expliqué en détails.

6.1 Les différents modules dans PersonFit

Deux modèles issus de l'architecture des STI sont implémentés dans PersonFit. Il s'agit du modèle des connaissances du domaine et du modèle de l'apprenant. La figure 6.1 présente ces deux modèles. Également sur cette figure il y a trois composants importants de l'architecture de PersonFit, à savoir:

- *QuestionRecommender*: qui est le mécanisme de sélection des items, d'adaptation des évaluations et du diagnostic cognitif de l'apprenant.
- *CurriculumRecommender*: qui est le mécanisme de recommandation de cheminement d'apprentissage. Il ne sera pas présenté dans ce mémoire.
- *Agent tuteur* qui communique avec l'apprenant, fait le diagnostic cognitif, etc.

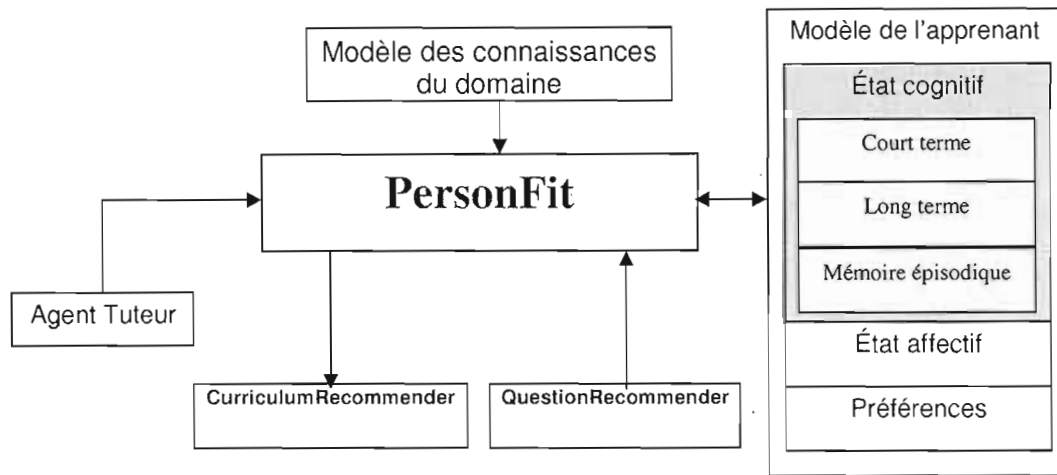


Figure 6.1. Architecture simplifiée des composants de PersonFit et leurs interactions

6.1.1 Modélisation des connaissances du domaine

La base des connaissances du domaine sera montée dans une perspective pédagogique (pré-requis, acquisition, etc.) correspondant à l'enseignement des concepts du domaine tout en prenant en compte des perspectives d'évaluation. De façon concrète, une définition de ces concepts et de leurs sous-concepts sera effectuée. Ceci servira de base pour créer les catégories (dimensions au sens de la TRI) qui permettront d'organiser les items. Dans les prochaines versions de PersonFit, ces concepts serviront de base pour une approche de représentation ontologique des connaissances du domaine. En guise d'exemple, la figure 6.2 représente les concepts de l'enseignement d'un cours de base de données. Les concepts sont représentés par les cercles. À l'intérieur de chaque cercle se trouve leur nom du concept associé. De chaque concept partent des flèches qui pointent vers les sous-concepts qui y sont associés. Le niveau de granularité est tributaire des objectifs visés et du concepteur.

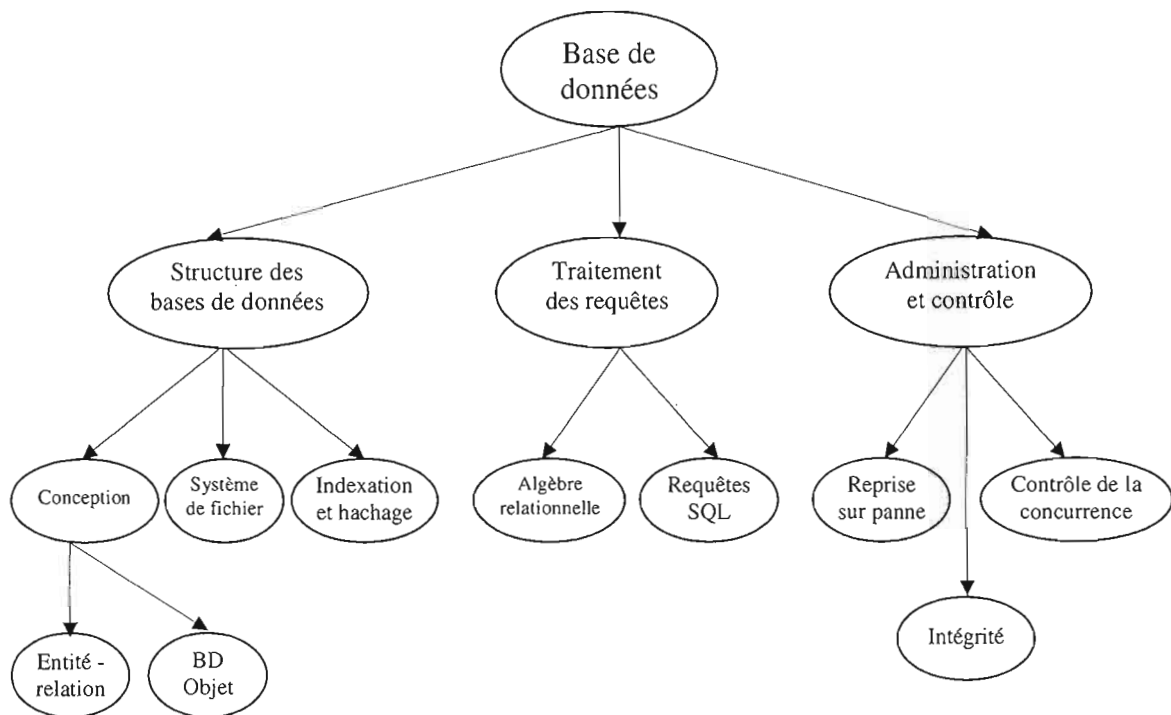


Figure 6.2. Modélisation des concepts pour un cours de base de données

6.1.2 La modélisation de l'apprenant

Le modèle de l'apprenant dans PersonFit comporte trois principales composants à savoir *le profil cognitif*, *l'état affectif* et les *préférences*. Ce mémoire présentera uniquement le profil cognitif. Ce profil comprend une représentation de l'état des connaissances de l'apprenant à court terme, une représentation de l'état des connaissances de l'apprenant à long terme et la mémoire épisodique.

L'état des connaissances à court terme

L'état des connaissances à court terme consigne principalement la valeur estimée du niveau d'habileté (θ), son erreur-type (S_θ) et un indice d'ajustement du patron de

réponses (L_z). Toutes ces données sont calculées pour la session d'évaluation courante et sont mises à jour à chaque réponse fournie par l'apprenant. Cet état est principalement utilisé par PersonFit pour la sélection des prochains items lors du déroulement de la session d'évaluation courante. Il est également utilisé par le module tuteur pour choisir la stratégie d'interaction avec l'apprenant ainsi que faire une détection éventuelle de jeu (*gaming*) chez ce dernier. La TRI sera utilisée comme modèle de calcul. À la fin de chaque session d'évaluation, les différentes données calculées ou inférées sont utilisées pour mettre à jour l'état des connaissances à long terme. Les données sont ensuite effacées de la mémoire et ne sont pas stockées.

L'état des connaissances à long terme

L'état des connaissances à long terme représente l'état de connaissances d'un apprenant tel qu'interprété au fil des résultats aux tests qui lui ont été administrés. Le modèle des connaissances à long terme dans PersonFit est un recouvrement des connaissances du domaine (section 6.1.2). Il exprime pour chaque concept du domaine de connaissance concerné leur probabilité d'acquisition. Ces connaissances à long terme servent principalement à l'initialisation du modèle de l'apprenant au début d'une session d'évaluation et à générer des rapports de diagnostics sur l'apprenant. Les RB sont utilisés comme modèle de représentation à cause des possibilités d'inférence et d'apprentissage qu'ils offrent. Chaque nœud du RB correspond à un concept du domaine des connaissances. À chaque concept est associée une probabilité issue de l'estimation du niveau d'habileté. Si l'apprenant subit une évaluation sur les mêmes concepts, les valeurs sont mises à jour avec la nouvelle valeur du niveau d'habileté. La provenance de chacune des informations contenues dans le réseau est conservée, à savoir s'il s'agit de données provenant d'observation directe des résultats de l'apprenant ou d'une inférence bayésienne à partir de ces données; dans ce cas, il s'agit d'une observation indirecte. L'état des connaissances à long terme sera physiquement stocké sous forme d'un fichier XML.

La figure 6.3 est une illustration d'un RB couvrant le domaine de l'enseignement des bases de données. Elle résulte du modèle des connaissances du domaine représenté par la figure 6.2. Cette figure montre la construction du réseau en utilisant le programme javaBayes¹¹. À chaque concept présent dans la modélisation des connaissances du domaine est associé nœud ou variable du RB. Les probabilités associées aux arcs qui relient ces nœuds sont initialisées par défaut par le tuteur. Mais, elles peuvent être estimées par la suite par apprentissage machine à partir des données. Ainsi, à partir de la valeur connue du niveau d'habileté d'un apprenant dans un concept, on peut, par inférence, prévoir la valeur du niveau d'habileté dans un autre concept appartenant au même RB.

La fenêtre dans l'angle supérieur droit de la figure 6.3 montre la table de probabilité correspondant au concept *QueryProcessing* ainsi qu'à ses sous-concepts notamment *SQLqueries* et *RelationalAlgebra*. À chacun des concepts peut être associé une taxonomie. Dans le présent cas, un sous-ensemble de la taxonomie de Bloom (Bloom *et al.*, 1956) correspondant à la connaissance (*Knowledge*) du concept, son application (*Application*) et son évaluation (*Evaluation*) a été utilisé. Le tableau 6.1 présente en ordre le nom des variables auxquelles sont associées les différentes probabilités.

¹¹ <http://www.cs.cmu.edu/~javabayes/>

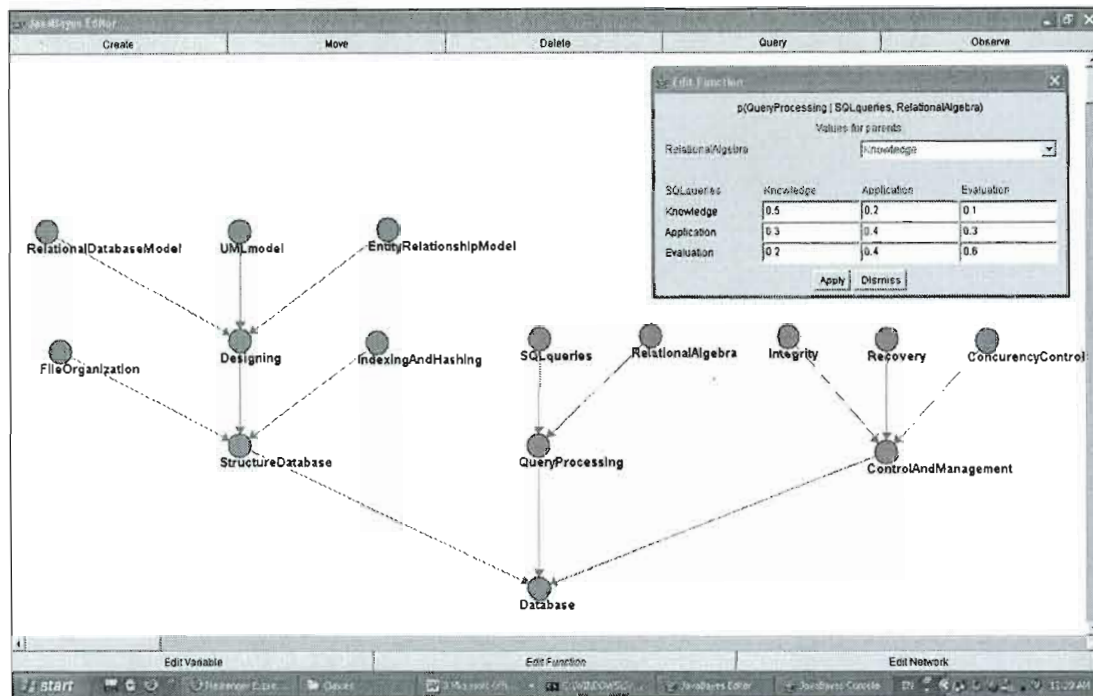


Figure 6.3 Structure d'un RB couvrant les concepts d'un cours de base de données

Les variables (A1... C3) décrites dans le tableau 6.1 correspondent respectivement aux différentes valeurs qui sont représentées dans la fenêtre de l'angle supérieur droit de la figure 6.3.

A1 = Knowledge du concept QueryProcessing	B1 = Knowledge du concept SQLQueries	C1 = Knowledge du concept RelationalAlgebra
A2 = Application du concept QueryProcessing	B2 = Application du concept SQLQueries	C2 = Application du concept RelationalAlgebra
A3 = Evaluation du concept QueryProcessing	B3 = Evaluation du concept SQLQueries	C3 = Evaluation du concept RelationalAlgebra

Tableau 6.1. Différents variables associées au concept de traitement des requêtes

La table de probabilité de $p(\text{QueryProcessing} | \text{SQLqueries}, \text{RelationalAlgebra}) = p(A|B,C)$ est représentée dans le tableau 6.2

Connaissance du concept A	Application du concept A	Évaluation du concept A
$p(A_1 B_1, C_1) p(A_1 B_1, C_2) p(A_1 B_1, C_3)$	$p(A_2 B_1, C_1) p(A_2 B_1, C_2) p(A_2 B_1, C_3)$	$p(A_3 B_1, C_1) p(A_3 B_1, C_2) p(A_3 B_1, C_3)$
$p(A_1 B_2, C_1) p(A_1 B_2, C_2) p(A_1 B_2, C_3)$	$p(A_2 B_2, C_1) p(A_2 B_2, C_2) p(A_2 B_2, C_3)$	$p(A_3 B_2, C_1) p(A_3 B_2, C_2) p(A_3 B_2, C_3)$
$p(A_1 B_3, C_1) p(A_1 B_3, C_2) p(A_1 B_3, C_3)$	$p(A_2 B_3, C_1) p(A_2 B_3, C_2) p(A_2 B_3, C_3)$	$p(A_3 B_3, C_1) p(A_3 B_3, C_2) p(A_3 B_3, C_3)$

Tableau 6.2. Table des probabilités correspondant à $p(A|B,C)$

Même si certaines valeurs de la table de probabilités représentée dans le tableau 6.2 ne sont pas connues, elles peuvent être estimées par inférence bayésienne. Par exemple, on peut estimer pour un apprenant la probabilité de connaître le concept A, s'il sait évaluer le concept B et connaît le concept C.

La mémoire épisodique

La mémoire épisodique emmagasine les traces des actions que l'apprenant a effectuées pendant son interaction avec PersonFit au cours d'une session d'évaluation. La trace d'évaluation est stockée au format XML à cause de la diversité de la nature et du type des informations qui y seront stockées. Son objectif est de permettre une sauvegarde physique des données jugées pertinentes lors d'une session d'évaluation à l'exclusion de l'état des connaissances de l'apprenant. Par exemple, on pourrait y retrouver l'identification des items qui ont été administrés à l'apprenant durant les diverses évaluations subis, les temps de réponse, etc. Ceci est

consulté par le mécanisme de sélection des items lors d'une évaluation afin d'empêcher de présenter le même item à un apprenant qui subit plusieurs évaluations dans un même domaine.

6.1.3 L'agent tuteur

Le tuteur dans PersonFit aura principalement pour fonction d'administrer des tests adaptatifs ou non aux apprenants, de consulter le modèle de l'apprenant, de détecter le jeu lors d'une session d'évaluation et de faire le diagnostic cognitif assez simple. En ce qui concerne le dernier point à savoir celui du diagnostic cognitif de l'apprenant, en principe ceci se fait dans le modèle des connaissances pédagogiques, car il est considéré comme une fonction pédagogique réalisée par un tuteur. Mais comme PersonFit n'intègre pas de modèle pédagogique, cette fonctionnalité a été ajoutée au composant jouant le rôle de tuteur. Ceci permet également d'avoir une architecture assez simple.

Dans PersonFit, selon la nature de l'évaluation (adaptative ou non), l'agent tuteur communique avec l'apprenant avant, durant et après une évaluation. Pour ce faire, dans la conception de l'interface de présentation des items aux apprenants, il y aura une fenêtre principale pour l'affichage des items à laquelle s'ajouteront des fenêtres secondaires dépendamment du type d'interaction.

6.2 Le *QuestionRecommender*

Les deux fonctions principales du *QuestionRecommender* consistent à sélectionner des items et les recommander à l'agent tuteur de PersonFit pour l'adaptation des évaluations. Une version simplifiée de l'algorithme utilisé par le *QuestionRecommender* est présentée dans l'organigramme de la figure 6.4. Ensuite suivra l'explication de quelques points importants de cet algorithme.

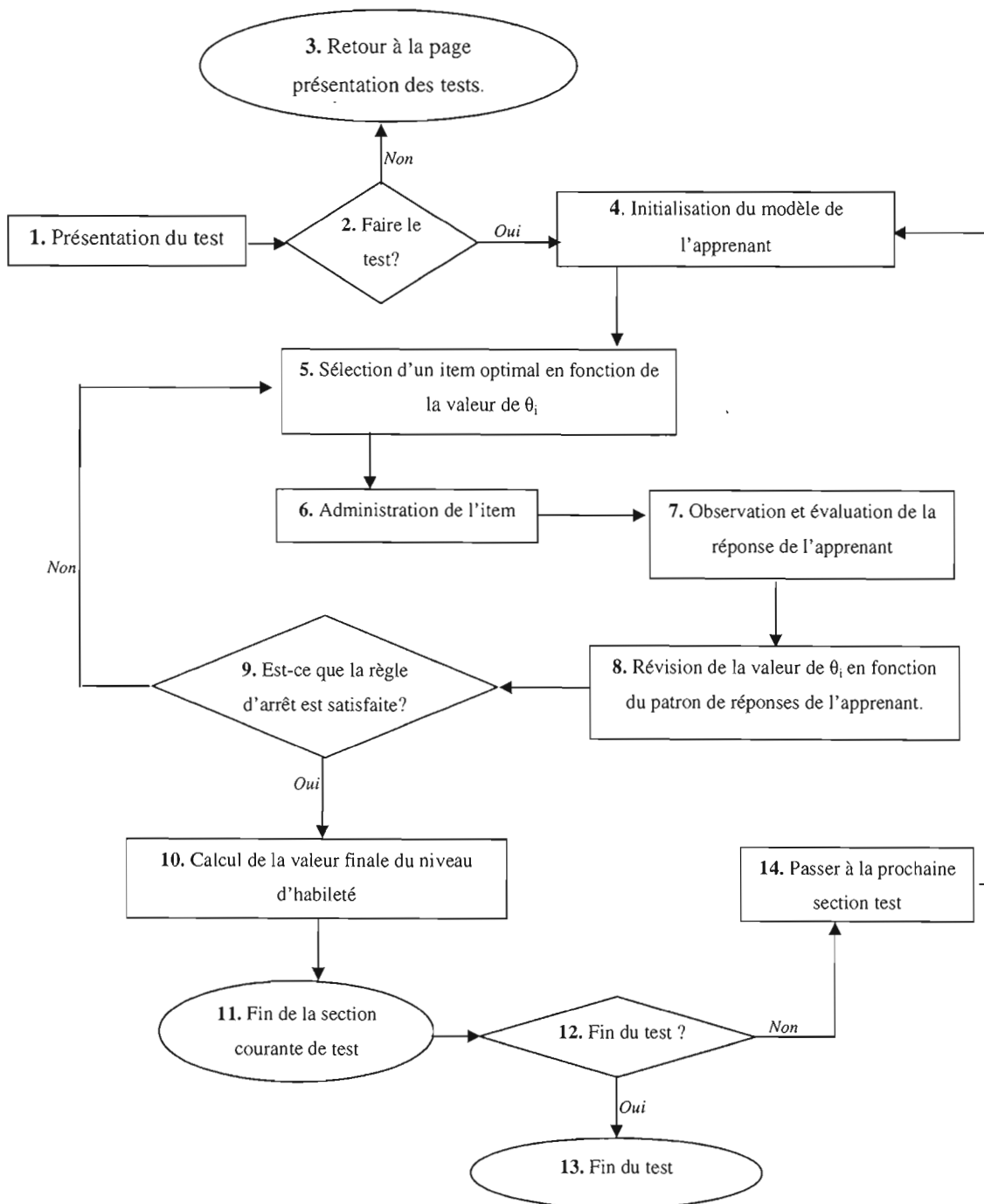


Figure 6.4. Organigramme simplifié de l'algorithme de test adaptatif

Explication de quelques points de l'algorithme

Comme mentionné, l'objectif de PersonFit est de reproduire le comportement qu'aurait un examinateur expérimenté. Ce dernier prendrait des décisions sur les questions à sélectionner et à poser à un apprenant afin d'obtenir le plus précisément possible son niveau d'habileté. Ainsi dans l'explication des différentes étapes de l'algorithme, un parallèle avec les décisions d'un examinateur humain sera effectué.

1. La présentation du test consiste à afficher des informations de nature descriptive sur le test. Par rapport à un examinateur humain, cette étape consiste à mettre l'apprenant dans le contexte de l'évaluation qu'il veut lui faire subir en lui donnant des informations comme la nature de l'évaluation, les objectifs visés, etc. Dans PersonFit, ces informations sont fournies au moment de la conception du test par son auteur. Au nombre de ces informations, nous pouvons citer le nom du test, les concepts sur lesquels portent le test, l'objectif visé par le test, le type de test, le nombre de questions, la description générale, la date de début et la date de fin du test, la durée, le nombre de questions, etc. Selon le type de test (adaptatif ou non), certaines des informations précitées sont optionnelles. Après avoir lu les informations, l'apprenant peut décider de faire le test. Ce qui l'amène à l'étape 2 de l'algorithme.

2. A cette étape, l'apprenant peut décider de faire ou non le test. S'il décide de faire le test, il débutera par la première section si le test est constitué de plusieurs sections.

4. Cette étape porte sur l'initialisation du modèle de l'apprenant. Elle a principalement pour objectif de déterminer de la façon la plus précise possible, une valeur initiale du niveau d'habileté de l'apprenant dans le domaine sur lequel porte l'évaluation; de même que la précision sur cette valeur. Pour un examinateur humain, cela consiste à estimer un niveau d'habileté initial de l'apprenant en se basant sur les

informations dont il dispose sur ce dernier. Même si cette valeur vient avec un certain doute sur sa précision ou son exactitude, tout doit être mis en œuvre pour obtenir une valeur la plus précise que possible.

Pour déterminer la valeur initiale du niveau d'habileté de l'apprenant, deux situations classiques se présentent. La première concerne un apprenant dont le niveau d'habileté est connu dans le domaine concerné par l'évaluation ou pour les concepts reliés à ce domaine. Dans ce cas, l'estimé initial du niveau d'habileté de l'apprenant est obtenu par inférences bayésiennes à partir des données à long terme sur l'état de ses connaissances. Dans le second cas, il s'agit d'un apprenant n'ayant reçu aucun test dans des concepts concernés par l'évaluation courante. La valeur initiale de son niveau d'habileté est obtenue en faisant la moyenne des scores obtenus lors des administrations précédentes de tests adaptatifs à d'autres apprenants du groupe d'où provient l'apprenant en situation de test. Cette valeur initiale assure que la difficulté du premier item administré soit égale au niveau d'habileté moyen de son groupe. L'item sélectionné permet également une discrimination optimale de l'apprenant par rapport à son groupe. En l'absence d'information à propos d'autres étudiants de son groupe, la valeur initiale du niveau d'habileté de l'apprenant est fixée à 0.

Il importe de souligner ici que la détermination de la valeur initiale du niveau d'habileté ($\hat{\theta}_{a \text{ priori}}$) peut affecter l'estimateur final du niveau d'habileté ($\hat{\theta}$) lorsque trop peu d'items sont administrés. C'est d'ailleurs la raison pour laquelle certains auteurs suggèrent d'utiliser la même valeur de l'estimateur a priori du niveau d'habileté pour tous les apprenants à qui est administré un test adaptatif. Étant donné que le $\hat{\theta}_{a \text{ priori}}$ a un impact sur la vitesse de convergence vers la valeur de $\hat{\theta}$ tout est mise en œuvre pour l'estimer de la façon la plus précise possible.

À la fin de l'étape l'initialisation du modèle de l'apprenant, la valeur initiale estimée du niveau d'habileté ($\hat{\theta}_{\text{a priori}} = \theta_i$ avec $i = 0$) de l'apprenant ainsi que l'erreur-type sur cette valeur (S_{θ}) sont connus. La valeur de l'indice d'ajustement du patron de réponses (L_z) est initialisée à 0. Ces trois informations constituent les données principales de l'état des connaissances à court terme de l'apprenant, qui seront mises à jour tout au long du test.

5. Cette étape porte sur la sélection d'un item optimal à administrer en fonction de la valeur de θ_i . Elle consiste à choisir l'item pour lequel l'information (section 4.4) est maximale pour le niveau d'habileté actuel calculé de l'apprenant: θ_i . Pour faire le parallèle avec un examinateur humain, cette stratégie consiste à poser au répondant une question dont la réponse fournie va permettre de maximiser l'information révélée par la valeur courante du niveau d'habileté. Étant donné qu'il serait coûteux et inefficace de parcourir toute la banque d'items du domaine pour chaque valeur de θ_i , ensuite calculer l'information fournie en vue de choisir l'item le plus informatif, une table obtenue par simulation est utilisée. Cette table porte le nom de *table d'information*. Elle contient une approximation de l'information fournie par chaque item de la banque d'items pour différentes valeurs du niveau d'habileté. La procédure de sélection en utilisant la table d'information consiste alors à choisir l'item qui fournit le plus d'information à une valeur rapprochée du niveau d'habileté. L'utilisation de *table d'information* constitue un énorme gain en temps d'exécution car elle permet un accès direct au lieu d'un accès séquentiel en parcourant toute la banque d'item. Elle est chargée en mémoire au démarrage du test; après avoir obtenu l'identification d'un item, on accède directement à son fichier source dans la banque d'items.

D'un point de vue technique, la *table d'information* contient les identifiants d'un nombre n d'items en ordre croissant de la valeur de l'information fournie à différents

intervalles de θ (Wainer, 2000). Le tableau 6.3 fournit une simple illustration d'une table d'information. Dans cette table on peut remarquer, par exemple, que pour un apprenant de niveau d'habileté -1, l'item 125 est le plus informatif. Par contre pour un apprenant de niveau d'habileté 1, le même item devient le moins informatif. Un autre avantage de la table d'information est de pouvoir déterminer la pertinence des items de la banque. Si un item ne se retrouve pas dans la table d'information, cela signifie que cet item n'est pas parmi les n items les plus informatifs pour tous les niveaux d'habileté.

θ	Item par ordre d'information fournie pour chaque valeur de θ				
-1	125	18	52	25	78
0	18	24	2	56	8
1	5	18	85	25	125

Tableau 6.3. Exemple fictif d'une table d'information

D'autres contraintes sont également prises en compte dans la sélection du prochain item. Il s'agit notamment de tenir compte d'une meilleure couverture du contenu pédagogique du domaine concernée et d'une minimisation de l'exposition de certains items qui, plus informatif, risquent d'apparaître plus souvent dans les tests (cas de l'item de numéro 18 dans le tableau 6.3).

Un composant de PersonFit nommé **Profiler** intervient également à cette étape. Le **Profiler** tourne en arrière-plan durant le test pour faire une analyse de la séquence des questions et tenter de bâtir un profil pour le répondant. À cette fin, il tente d'établir certaines évidences qui peuvent être ajoutées dans son état de connaissance à long terme. Si la réussite ou l'échec d'un item permet de confirmer une hypothèse du Profiler, cet item devient prioritaire et est sélectionné comme item optimal.

7. Deux actions sont réalisées l'une après l'autre à cette étape. La première est l'observation. Elle consiste à observer et enregistrer les renseignements concernant

les actions du répondant, avant, durant et après avoir donné sa réponse à l'item qui lui est présenté. Il s'agit d'informations comme le temps de réponse, les changements effectués à la réponse donnée, etc. La seconde action porte sur l'évaluation. Elle permet d'attribuer une valeur de succès (1) ou d'échec (0) à la réponse finale fournie par le répondant. Ces deux actions pour un examinateur humain consistent à observer chez l'apprenant évalué le doute, la frustration, les changements de réponses, etc. Par la suite, de déterminer si la réponse donnée est correcte ou non.

8. Cette étape porte sur la révision de l'estimateur de la valeur de θ_i en fonction du patron de réponses de l'apprenant. Également, la précision (S_θ) et la valeur de l'indice d'ajustement du patron de réponses (L_z) sont recalculées. Après cette révision, l'indice i est incrémenté et θ_{i+1} correspond à la nouvelle valeur calculée de θ . Pour un examinateur humain, cela consiste à réévaluer l'estimation qu'il avait du niveau d'habileté de l'apprenant en prenant en considération la réponse fournie à l'item courant qui lui a administré.

Un composant de PersonFit nommé *ThetaEstimator* se charge de ces opérations. Pour l'estimation de la valeur du paramètre du sujet (θ), une version adaptative de la méthode MAP (AMAP) est utilisée. Les valeurs estimées du niveau d'habileté après n items y sont utilisées dans le but d'estimer la nouvelle valeur à l'étape $n+1$. La méthode AMAP tient donc compte de l'information disponible sur l'apprenant a priori dans les calculs des étapes subséquentes; de plus, elle offre de bons rendements au regard du temps de calcul et de la précision de l'estimateur.

Par rapport à la nouvelle valeur de l'indice d'ajustement du patron de réponses (L_z), on procède également à cette étape à la détection de jeu.

9. Cette étape porte sur la règle d'arrêt du test ou d'une section du test. Deux méthodes sont généralement utilisées. La première qui est la plus utilisée, consiste à

arrêter le test lorsqu'une valeur prédéterminée de l'erreur-type de l'estimateur du niveau d'habileté (S_θ) est obtenue. Cette règle d'arrêt permet d'obtenir environ la même erreur-type à tous les niveaux d'habileté estimés et donc pour tous les apprenants. En utilisant cette règle d'arrêt, une solution est donc apportée au problème de l'hétérogénéité de la variance de l'estimateur du niveau d'habileté pour les tests classiques. Mais il pourrait y avoir un problème de non-convergence θ . Ce problème survient généralement lorsqu'il y a une grande variabilité dans le patron de réponse de l'apprenant. Ceci est à la base de la seconde règle d'arrêt utilisée. Elle consiste à terminer le test après un nombre préalablement fixé d'items.

Une situation particulière pourrait se présenter dans le cas où il n'existe pas beaucoup d'items dans la banque. C'est que parmi les items restants, aucun ne maximise l'information fournie par la valeur courante du niveau d'habileté. Cette situation peut être évitée par la création d'une grande banque d'items. Mais si elle se présente tout de même, mieux vaut mettre fin au test.

Tant que la règle d'arrêt n'est pas satisfaite, un nouvel item plus informatif par rapport à la valeur courante de θ_i et respectant les diverses contraintes est sélectionné et administré. Le processus itératif portant sur les étapes 5 à 9 se poursuit, jusqu'à ce que la règle d'arrêt soit satisfaite.

10. Un test peut être constitué de plusieurs sections portant chacune sur différents aspects. Si tel est le cas l'algorithme reprend avec une nouvelle section, comme si un nouveau test était administré.

Sommaire sur le déroulement d'un test adaptatif en utilisant la TRI

En se basant sur les trois règles descriptives des tests énoncés (section 2.2) et sur la stratégie d'adaptation des évaluations utilisée, nous obtenons le tableau suivant décrivant le déroulement d'un test adaptatif basé sur la TRI.

RÈGLES	ACTION
1. Règle de départ	Initialiser le modèle de l'apprenant en tenant compte de l'information disponible a priori ou qu'on peut inférer sur l'apprenant.
2. Règle de suite	Administrer un item dont les paramètres permettent de maximiser l'information fournie selon la valeur courante du niveau d'habileté. Ceci se fait en tenant compte de certaines caractéristiques du profil du répondant.
3. Règle d'arrêt	Terminer le test après l'administration d'un nombre prédéterminé d'items, lorsqu'une erreur-type prédéterminée de l'estimateur du niveau d'habileté est obtenue ou lorsqu'il n'y a plus d'items qui puissent fournir une quantité d'information maximale au niveau d'habileté estimé.

Tableau 6.4. Déroulement d'un test adaptatif basé sur la TRI

6.3 Conclusion du chapitre

Le chapitre VI nous a permis de faire une description détaillée du principe mis en œuvre par PersonFit pour l'adaptation des évaluations. Les différentes modélisations utilisées ont été également présentées. L'algorithme de déroulement de test adaptatif utilisé est présenté et expliqué en détails. À l'étape 8 de ce algorithme, nous retrouvons la détection de jeu pour s'assurer de la cohésion du patron de réponse de l'apprenant par rapport à son niveau d'habileté et les réponses fournies aux items qui lui ont été administrés. Ceci est préalable au diagnostic cognitif qui sera effectué plus tard et qui utilise les résultats aux tests, car il permet de s'assurer que l'apprenant ne joue pas avec la plate-forme. Ces deux fonctionnalités, à savoir la détection et le diagnostic cognitif, ne constituent pas les objectifs premiers de développement de PersonFit. Elles ne seront présentées que succinctement dans le prochain chapitre.

CHAPITRE VII

DIAGNOSTIC COGNITIF ET DÉTECTION DE JEU DANS PERSONFIT

Le chapitre VII permet de présenter séparément et de façon succincte deux fonctionnalités connexes développées dans PersonFit. Il s'agit de la détection de jeu à partir des patrons de réponses inappropriés et le diagnostic cognitif des apprenants.

7.1 Le diagnostic cognitif de l'apprenant

7.1.1 Le concept de diagnostic cognitif dans les STI

De par son étymologie, le terme "diagnostic" signifie "connaissance à travers les signes". Ainsi, dans un STI, la forme la plus primitive du diagnostic cognitif consiste selon Wenger (1987), à reconnaître, caractériser, classer et juger les actions observables de l'apprenant. Le diagnostic cognitif peut donc être interprété comme la connaissance de l'état cognitif de l'apprenant à travers les signes (actions, réponses aux questions, réactions lors d'une activité d'apprentissage, etc.) qu'il présente. Il existe différents types de diagnostic cognitif de l'apprenant (Tchetagni, 2005). Dans PersonFit, c'est le diagnostic épistémique qui a été implémenté. Par la suite dans ce

mémoire, l'expression 'diagnostic cognitif' fera référence à ce type de diagnostic. Le choix du diagnostic épistémique se justifie par son principe de fonctionnement assez simple. Il consiste à inférer l'état des connaissances de l'apprenant qui pourrait expliquer sa performance lors de la résolution d'une question ou d'un problème. En prenant comme exemple le diagnostic médical qui fait l'association entre les symptômes et les causes, dans le contexte d'apprentissage, il fait associer le comportement observable de l'apprenant pendant une activité d'évaluation à l'état cognitif qui expliquerait le mieux ce comportement ou ce résultat. Cependant, le diagnostic épistémique présente une grande faiblesse. Il ne permet pas d'inférer le raisonnement qui sous-tend la génération des réponses qu'il analyse. Or, cet aspect est très important dans certaines disciplines comme les mathématiques, etc.

Le processus générique utilisé pour un diagnostic cognitif dans un STI, peut être divisé en quatre étapes suivantes (Tchetagni, 2005) :

Étape 1: L'acquisition des indices. Cette acquisition doit être contrôlée par les prémisses qui ont été préalablement fixées sur le genre d'inférence voulu.

Étape 2: La génération d'hypothèses par inférence et l'interprétation de ces hypothèses se font à cette étape.

Étape 3: Le diagnostic proprement dit est effectué à cette étape. Il consiste à établir par le biais de tests adéquats quelles sont les hypothèses de l'étape 2 qui sont valides. Cela permet de déterminer les hypothèses qui seront effectivement confirmées comme résultat final du diagnostic.

Étape 4: L'ajustement des résultats du diagnostic afin d'être utilisé pour orienter les stratégies des prochaines acquisitions d'indices.

Ces différentes étapes peuvent être exécutées de façon cyclique. Suite aux différentes étapes présentées ci-dessous, la prochaine section présentera l'implémentation du diagnostic cognitif dans PersonFit.

7.1.2 Implémentation du diagnostic cognitif dans PersonFit

Pour l'implémentation du diagnostic cognitif dans PersonFit, les comportements observables retenus sont les réponses fournies par l'apprenant aux items qui lui ont été administrés, les temps de réponse, etc. Ces données observées seront donc utilisées comme les indices pour établir le diagnostic. Toutefois, cette implémentation sera assez délicate. En effet, dans les STI classiques, le diagnostic est souvent basé sur une banque des erreurs les plus courantes et les connaissances reliées à ces erreurs. Cette approche est adaptée au STI, car ils sont généralement utilisés comme support pour l'enseignement dans un domaine précis. Étant donné que PersonFit n'est pas conçu pour un domaine précis, la conception de la banque des erreurs sera alors difficile. Le mécanisme de diagnostic utilisé se base sur celui des STI. Ensuite, des ajustements ont été apportés pour obtenir une approche basée sur la prescription. Ceci consiste à observer un échec et voir les pré-requis qui pourraient en être la cause.

La figure 7.1 est une illustration de la boucle de fonctionnement de la stratégie diagnostic et de remédiation utilisée. Dans les prochaines sections, le détail de chacun des points de figure 7.1 sera présenté.

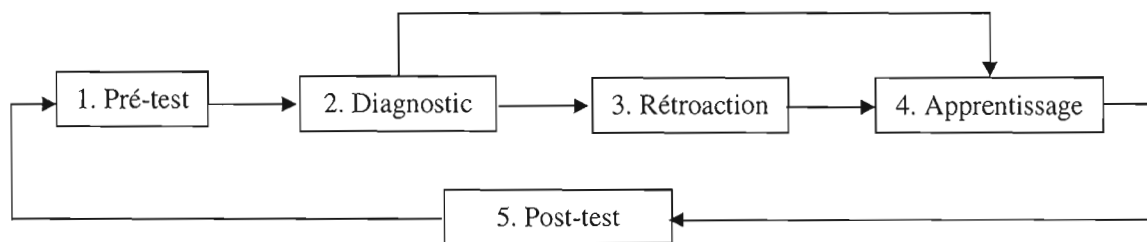


Figure 7.1 Boucle de diagnostic cognitif et de remédiation dans PersonFit

1. Le Pré-test

Il s'agit d'un test sur mesure initié par PersonFit pour rechercher, identifier ou confirmer les causes de certaines mauvaises réponses fournies par l'apprenant dans

d'autres tests. Étant donné que la mémoire épisodique et le modèle de connaissance à long terme d'un apprenant sont stockés physiquement, le composant **Profiler** peut procéder à l'analyse de leurs contenus. Ceci permettra d'identifier les concepts sur lesquels portent les items auxquels l'apprenant n'a pas bien répondu. Le **Profiler** suppose que l'origine des échecs se trouve au niveau des concepts identifiés ou de leurs pré-requis. Le pré-test va donc porter principalement sur ces pré-requis et peut être adaptatif ou non. Également, si pour un concept les items faussés sont d'un haut niveau taxonomique, dans le pré-test le niveau taxonomique des items administrés sera abaissé.

En guise d'exemple considérons la figure 7.2 ci-dessous qui représente un extrait de la figure 6.2 des concepts pour l'enseignement d'un cours de base de données. Si un apprenant donne une mauvaise réponse à plusieurs questions relatives au concept *Traitement des requêtes*, le pré-test va principalement porter sur les deux concepts pré-requis associés que sont *Algèbre relationnelle* et *Requêtes SQL*.

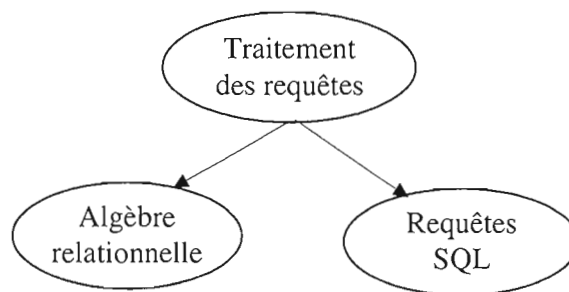


Figure 7.2 Extrait du modèle des connaissances du domaine

Par rapport au processus générique de diagnostic cognitif (section 7.1.1), le pré-test peut être perçu comme la phase d'acquisition d'indices. Les résultats au pré-test vont donc permettre de procéder au diagnostic cognitif proprement dit.

2. Le diagnostic cognitif

Le résultat du pré-test a une influence directe sur le résultat du diagnostic. Il permet de poser les hypothèses pour tenter d'identifier les causes possibles de ce résultat. Par rapport au processus générique de diagnostic cognitif, ceci peut être perçu comme la phase de génération d'hypothèses. À cette étape, certains cas très simples peuvent se présenter. Par exemple, si l'apprenant trouve la bonne réponse à la majorité des items associés au concept Requêtes SQL mais que le contraire se produit pour le concept *Algèbre relationnelle*, on considère tout simplement comme hypothèse la méconnaissance du concept *Algèbre relationnelle*. Mais les choses ne sont pas toujours aussi simples. Ainsi, il peut arriver que les items auxquels l'apprenant a mal répondu ne soient pas tous associés à un concept même. Si tel est le cas, le RB sous-jacent à l'état des connaissances à long terme de l'apprenant est mis à contribution.

La théorie des RB fournit des fondements pour identifier l'état le plus probable qui pourrait expliquer une observation, ceci se fait généralement par une *abduction bayésienne* (section 4.6.2). Comme exemple très simple, considérons la figure 7.3. Supposons que θ_1 (la valeur estimée du niveau d'habileté) est observé. On peut déterminer la valeur de vraisemblance de θ_2 et θ_3 qui permettent d'obtenir cette valeur de θ_1 . Ces résultats sont comparés à ceux obtenus dans le pré-test. Ensuite, l'algorithme MPE (*Most Probable Explanation*) est utilisé pour délivrer une interprétation des échecs de l'apprenant (Nkambou et Tchetagni, 2004). Bien entendu, le diagnostic est plus précis et plus facile quand plusieurs évidences sont disponibles plutôt que quelques constats d'échec ou de succès aux items.

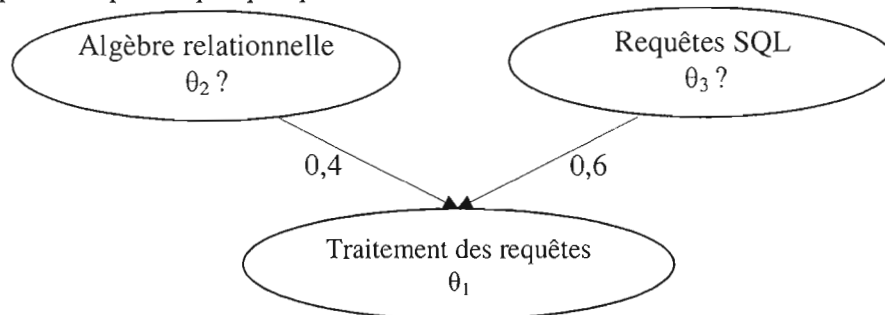


Figure 7.3 Estimation de la valeur du niveau d'habileté des pré-requis

Pour faciliter l'acquisition des indices pour le diagnostic cognitif, il arrive que le concept d'*item clé* soit utilisé. Il s'agit d'items dont la réussite ou l'échec sont très informatifs et sont considérés comme fondamentaux par le tuteur. En cas d'échec aux items clés associés à un concept, les évidences associées sont automatiquement prises en considération pour le diagnostic.

3. Rétroaction

Il s'agit d'une rétroaction générée par PersonFit. Elle sera présentée sous forme d'une fiche détaillant de façon sommaire les résultats du diagnostic cognitif de l'apprenant. Les fiches seront générées sur la base d'un gabarit. Les données proviennent de la base de données, du modèle de connaissance à long-terme de l'apprenant et des résultats du diagnostic cognitif de l'apprenant. Un exemple de fiche est présenté dans le tableau 7.1. Ces fiches sont disponibles pour le tuteur. En consultant la fiche de diagnostic, le tuteur peut en cliquant sur des liens appropriés visualiser les informations suivantes :

- le détail des items administrés ainsi que leurs séquences au moment du test,
- les réponses fournies par le répondant,
- le contenu de l'état des connaissances à court terme,
- la décision ayant motivé la sélection de chaque prochain item administré.

Le tuteur peut autoriser un apprenant à les voir sa fiche diagnostic. Dans le cas contraire, l'apprenant passe directement de l'étape 2 à l'étape 4 de la boucle du diagnostic cognitif (figure 7.1).

Fiche de diagnostic			
<i>Document généré le lundi 20 juillet 2007 à 14h</i>			
Nom de l'apprenant: <i>Komi Sodoké</i>			
Cours : <i>Introduction aux bases de données relationnelles</i>			
Date du diagnostic: <i>12 mars 2007</i>			
Concepts concernés: <i>Traitement des requêtes</i>			
Résultat des évaluations			
	Évaluation1	Évaluation2 (Pré-test)	Évaluation3 (Post-test)
Date début	11/07/2007	13/07/2007	17/07/2007
Temps utilisé	15 min	20 min	17 min
Score	$\theta = -0,5$ $S_{\theta} = 0,01$	$\theta = 0,6$ $S_{\theta} = 0,01$	$\theta = 1,5$ $S_{\theta} = 0,01$
Jeu détecté	Non	Non	Non
<p>Raison du pré-test : Confirmer si les causes des mauvaises réponses obtenues par l'apprenant à l'évaluation1 sur le <i>Traitement des requêtes</i> proviennent du concept <i>Algèbre relationnelle</i> ou du concept <i>Requêtes SQL</i>.</p> <p>Résultat du diagnostic : L'analyse du patron de réponses de l'apprenant montre que l'apprenant a bien répondu à 7 des 9 questions portant sur le concept <i>Requêtes SQL</i>. L'apprenant a mal répondu à 5 des 6 questions portant sur le concept <i>Algèbre relationnelle</i>. Étant donné que ces deux concepts, sont les pré-requis du concept <i>Traitement des requêtes</i> sur lequel porte l'Évaluation1, PersonFit considère que la raison du mauvais résultat est : « mauvaise connaissance de: <i>Algèbre relationnelle</i> ».</p> <p>Recommandation effectuée : Revoir les sections du cours portant sur l'<i>Algèbre relationnelle</i>. [Cliquer ici pour y accéder].</p>			

Tableau 7.1 Exemple simplifié de fiche de diagnostic cognitif d'un apprenant

4. Apprentissage

Cette étape constitue le mécanisme de remédiation dans PersonFit après avoir identifié les causes des erreurs à l'étape 2. Elle consiste à prescrire des chapitres ou sections de chapitre d'un cours en ligne dans Moodle qui peuvent permettre à l'apprenant de réviser les concepts qui ont été identifiés lors du diagnostic cognitif comme sources des erreurs de l'apprenant. Le mécanisme nommé *CurriculumRecommender* a pour rôle la recommandation du cheminement d'apprentissage. Son fonctionnement ne sera pas présenté dans ce mémoire.

5. Le post-test

Cette étape constitue un mécanisme de validation. Elle permet de valider les apprentissages de l'apprenant après l'étape 4. Pour ce faire, on évalue la différence entre la valeur du niveau d'habileté au pré-test et celle au post-test. Le post-test est un test adaptatif. Il utilise les résultats de pré-test pour l'initialisation du modèle de l'apprenant. L'algorithme de tests adaptatifs présenté dans la section 6.5 est ensuite exécuté. Bien entendu, la mémoire épisodique est consultée pour ne pas présenter à l'apprenant des items déjà administrés au pré-test. Le post-test peut servir de pré-test à un potentiel prochain post-test. Ceci signifie que toutes les étapes décrites dans cette section peuvent s'exécuter de façon cyclique.

L'autre fonctionnalité connexe offerte par PersonFit est la détection du jeu. Son fonctionnement sera présenté dans la prochaine section.

7.2 La détection de jeu dans PersonFit

7.2.1 Le problème de jeu dans les STI

Le terme '*jeu*' (traduction du terme Anglais *gaming*) désigne l'ensemble des stratégies utilisées par des apprenants pour réussir dans un système éducatif en exploitant les faiblesses du système plutôt que d'étudier le matériel éducatif présenté pour ensuite utiliser la connaissance acquise afin de répondre aux questions. On pourrait également y associer des stratégies utilisées pour déjouer les objectifs des systèmes éducatifs. Le problème de détection de jeu est actuellement une grande préoccupation dans les systèmes éducatifs, y compris les STI (Baker *et al.*, 2006).

Dans PersonFit, nous aurons à faire face au problème de jeu principalement pour deux raisons. Premièrement, l'algorithme de déroulement de tests adaptatifs utilisé tend à administrer des items plus faciles après qu'un répondant ait mal répondu à des items subséquents plus difficiles ou vice versa. Comme conséquence, un apprenant astucieux pourrait employer de tels indices pour détecter des réponses correctes ou incorrectes aux items. Également, il pourrait donner délibérément de mauvaises réponses aux items, le menant ainsi à un test plus facile. Ceci est à l'origine de la deuxième raison de jeu qui est la volonté de sous-classement en vue de déjouer le mécanisme de diagnostic cognitif.

Dans la plupart des systèmes, Baker et ses collègues (Baker *et al.*, 2004) ont déterminé que le jeu pouvait être divisé en deux comportements distincts qui sont :

1. le jeu « nocif », qui se produit généralement aux étapes de résolution de problème nécessitant des concepts que l'étudiant ne connaît pas. Il est donc associé au fait que l'apprenant veut éviter de pauvres résultats d'évaluation. L'une des causes majeures de jeu « nocif » chez les apprenants est la faible connaissance dans le domaine concerné par l'évaluation combinée avec la difficulté du problème.
2. le jeu « non-nocif », par opposition au jeu « nocif » se produit typiquement sur des étapes de résolution de problème ou sur des questions nécessitant des

concepts que l'étudiant connaît déjà ou maîtrise assez bien. Ce jeu est donc associé à la distraction ou la volonté de déjouer le système.

Dans PersonFit, c'est le deuxième cas, soit le jeu « non-nocif » qui retiendra le plus notre attention. Ceci se justifie par le fait que le jeu « nocif » serait très difficile à contrôler dans environnement distribué et sans supervision comme c'est le cas du E-testing. Par exemple, un apprenant pourrait tout simplement relire une section du cours concerné par une question pour ensuite donner la bonne réponse. En plus, comme l'objectif des tests administrés n'est pas certificatif, nous estimons que les tentations pour un apprenant de vouloir tricher ou d'utiliser d'autres moyens en vue d'augmenter son score seront faibles.

Afin de pallier les problèmes de jeu dans les systèmes éducatifs, différentes approches ont été utilisées. Nous allons présenter dans la prochaine section les approches utilisées dans les STI pour terminer par la celle utilisée dans PersonFit.

7.2.2 Les approches de solutions au jeu dans les STI

Ces approches sont principalement basées sur une analyse du comportement de l'utilisateur dans l'environnement du STI. Par exemple, des indicateurs comme le nombre d'aides demandées par l'apprenant pendant la résolution d'un problème, le temps de réponse, etc. sont utilisés (Baker et al., 2004). Également, différents attributs comme le nombre de tentatives de réponses, la durée de chaque tentative, la difficulté du problème, ainsi qu'une combinaison de ses facteurs sont aussi utilisés.

Le problème est que ces approches de solution ne sont pas basées sur le modèle utilisé pour la représentation de l'apprenant. Également, les différents indicateurs utilisés sont subjectifs et ne garantissent pas de faire des conclusions objectives. Par

exemple, un apprenant peut répondre rapidement aux questions ou activités qui lui sont proposées tout simplement parce qu'il est pressé et non qu'il tente de jouer avec le STI.

Somme toute, nous estimons qu'un ensemble de comportements observés chez un apprenant ne suffit pas pour tirer des conclusions évidentes de jeu chez ce dernier. Selon nous, une méthode plus objective pouvant être ajoutée à l'approche actuelle de détection du jeu dans les STI serait l'analyse de la cohérence des réponses fournies par l'apprenant. Ceci se fera pour chacun des items du test et aussi vérifier l'ajustement global du patron de réponse de l'apprenant par rapport à la valeur estimée de son niveau d'habileté. Pour ce faire, dans PersonFit, comme approche pour détecter le jeu, en plus de certains indicateurs précédemment cités; la contribution principale a été l'utilisation d'un indice. Cet indice permet de vérifier la cohérence des réponses fournies par l'apprenant aux items au regard de la valeur estimée de son niveau d'habileté ainsi que l'ajustement de son patron de réponse par rapport au modèle. Comme principe pour la vérification de la cohérence des réponses, nous nous basons sur le fait que les conséquences du jeu se traduiront par des patrons de réponses inappropriés, plutôt étranges et ainsi peu probables. Un indice issu de la TRI, qui est le modèle utilisé pour l'estimation des habiletés cognitives des apprenants, sera utilisé. Il s'agit de l'indice Lz.

7.2.3 L'indice Lz

Des réponses erronées ou volontairement biaisées par un apprenant ont une incidence sur l'estimation du niveau d'habileté du répondant concerné. Parmi les indices classiques dans la littérature sur la TRI, l'indice Lz a été retenu dans PersonFit pour trois raisons :

- 1) c'est un indice approprié à la plupart des modélisations issues de la TRI;

- 2) il est parmi les indices les plus utilisés dans la littérature et son comportement statistique est assez bien connu;
- 3) il donne d'assez bons résultats de détection.

7.3 Conclusion du chapitre

Ce chapitre nous a permis de présenter de façon succincte deux fonctionnalités connexes développées: la détection de jeu à travers les patrons de réponses inappropriés et le diagnostic cognitif des apprenants. Il marque la fin des chapitres décrivant les fonctionnalités de PersonFit et leurs principes de fonctionnement. Le prochain chapitre portera sur les stratégies d'implémentations utilisées.

CHAPITRE VIII

IMPLÉMENTATION DE PERSONFIT

Ce chapitre sera consacré à l'implémentation de PersonFit. Il présentera les choix technologiques effectués pour les différentes implémentations. Cette présentation du fonctionnement se fera avec quelques captures d'écran à l'appui.

8.1 Les langages de programmation et systèmes utilisés

Pour les implémentations, les deux langages de programmation les plus utilisés sont : le langage PHP et Java. Ces deux langages seront présentés dans les deux prochains paragraphes.

Le nom PHP¹² est un acronyme récursif pour *Hypertext Preprocessor*. PHP a été développé initialement par Rasmus Lerdorf. En 1997, PHP devient un projet collectif et est réécrit par Zeev Suraski et Andi Gutmans. C'est un langage de script qui fonctionne du côté serveur. Comme il supporte tous les standards du web et qu'il est gratuit, il s'est rapidement répandu sur la toile. Le langage continue d'évoluer. Actuellement, il est à la version 5. La majorité des serveurs y compris celui du laboratoire GDAC qui est utilisé pour nos expérimentations fonctionnent avec la

¹² <http://www.php.net>

version 4.x de PHP. Des principes de codage supportés aussi bien par la version 4.x que la 5 ont donc été adoptés. Des essais de ré-écriture des codes permettant l'estimation du niveau d'habileté en PHP ont été effectués en vue d'avoir un langage de développement unique du côté du serveur. Mais les résultats n'ont pas été très concluants quant à la précision et au temps de calcul. Comme solution, un couplage du PHP avec le langage C a été adopté. En effet, étant développé en C, le noyau de PHP offre des possibilités d'appeler, de passer des paramètres et d'obtenir les résultats directement d'un programme écrit en C. Ceci se fait sans passer par des fichiers tampons externes.

Quant au langage Java, il trouve ses origines dans les années 1990. À cette époque, des ingénieurs de SUN Microsystems¹³ ont commencé à envisager un environnement indépendant du matériel. James Grosling, un de ses ingénieurs, développa un premier langage nommé Oak permettant de programmer dans cet environnement. Il est renommé Java en 1995 et est soumis à la communauté Internet grandissante. Une machine virtuelle, ainsi que de nombreuses spécifications, sont ajoutées. Java entame alors une conquête fulgurante principalement comme solution à l'hétérogénéité des machines utilisées sur Internet. Aujourd'hui, après de nombreuses améliorations, Java n'est plus uniquement une solution liée à Internet. De plus en plus de sociétés et de particuliers utilisent ce langage pour toutes sortes de développements. Même si la technologie Java est libre et gratuite, elle demeure une propriété de Sun Microsystems. Le langage Java a été principalement utilisé pour le développement de l'application cliente de PersonFit.

Le tableau 8.1 présente les différents systèmes et programmes (C et PHP) qui interviennent dans le déroulement d'un test adaptatif au niveau du serveur et du navigateur client. Moodle est écrit en PHP et utilise une base de données MySQL. Le

¹³ <http://www.sun.com/>

modèle de l'apprenant est stocké en XML. Des programmes écrits en PHP permettent d'y accéder et de le mettre à jour.

Navigateur de l'apprenant (Interface étudiant de Moodle)	Serveur (Moodle, Code C, MySQL et Modèle de l'apprenant)
<div data-bbox="300 552 710 1602"> <pre> graph TD A[1. Identification du répondant] --> B[2. Mise à jour des informations sur le répondant. Date de visite, etc.] B --> C[3. Affichage des consignes du test] C --> D[4. Début du test et affichage de la première question] D --> E[5. Présentation de la première question et réponse de] E --> F[6. Seconde question] F --> G[7.] G --> H[8. Dernière question] </pre> </div>	<p>MySQL : Vérification du nom utilisateur et mot de passe.</p> <p>MySQL : mis à jour du profil de l'apprenant.</p> <p>MySQL : Sélection des informations sur le test.</p> <p>Modèle de l'apprenant : Initialisation ($\theta_{initial}$) Consultation de la table d'information pour sélectionner le premier item en fonction de la valeur initiale de θ et de S_θ</p> <p>Moodle: Formulaire de présentation de l'item. MySQL : Ajout de la réponse à la base de données Routine C : calcul de la valeur courante θ, S_θ et I_z. Modèle de l'apprenant : Mise à jour de l'état des connaissances à court terme.</p> <p>} Items subséquents. Répéter les actions du côté serveur de l'étape 5</p> <p>Moodle: Formulaire de présentation des items. MySQL : Ajout de la réponse à la base de données Routine C : Calcul de la valeur finale θ, S_θ et I_z. Modèle de l'apprenant : Ajout de l'identifiant des items administrés à la mémoire épisodique. Mis à jour de l'état des connaissances à long terme.</p>

Tableau 8.1. Intervention des différents systèmes dans un test adaptatif

8.2 Le format et la structure des items

8.2.1 Le format de stockage des items et paramètres d'item

Les items sont représentés par des fichiers XML au format IM-QTI 2. Les paramètres d'items sont également au format XML. Il n'existe pas de standard actuel en E-learning permettant la représentation des paramètres d'item. Comme solution, une structure propre au fichier source XML des paramètres d'items a été définie. Cette structure doit satisfaire certaines exigences. Par exemple, un même item peut servir pour l'évaluation de différents concepts et avoir des paramètres différents selon chaque concept. Également, aux fins des développements futurs, il peut arriver que différentes modélisations puissent être définies pour un même item. Le fichier des paramètres d'items est donc un fichier autonome dont un exemple de contenu est présenté dans le tableau 8.2.

```
<itemParameters    identifier=" param_question6">
    <model id="4PL"          dimension="concept_Id">
        <paramDetails a="0.1"      b="0.4"      c="0.2"      d="1" />
    </model>
</itemParameters>
```

Tableau 8.2. Exemple de fichier de paramètre d'item.

L'attribut *identifier* représente l'identifiant du fichier. Il est constitué de la chaîne de caractère " param_ " concaténée avec l'identifiant de la question dont les paramètres d'items sont présentés dans le fichier.

L'élément *model* représente le corps du fichier XML. L'attribut *id* permet de spécifier la modélisation utilisée. Dans cet exemple, l'attribut *id* a la valeur de "4PL", il s'agit donc d'un modèle dichotomique à 4 paramètres. L'attribut *dimension*

permet de spécifier l'identifiant du concept auquel correspond le détail des paramètres. Si le même item sert à évaluer plusieurs concepts, l'élément *model* est répété pour chaque concept en attribuant les valeurs correspondantes à l'attribut *dimension*. À l'intérieur de l'élément *model*, il y a l'élément : *paramDetails* dont les attributs permettent de spécifier les paramètres d'items selon la modélisation ainsi que leurs valeurs.

8.2.2 La structure de données pour la représentation des items

La figure 8.1 est une illustration de la structure de données utilisée pour représenter les items dans les programmes. Elle représente également la façon dont PersonFit aura à manipuler les items. Nous avons adopté le nom '*Assessment Object*' pour désigner cette structure. Comme traduction, nous utilisons le terme d'Objet d'Évaluation (OE). Par analogie au concept d'objet dans la programmation orientée objet, un OE est constitué à l'interne de deux composants:

- Le contenu : c'est un ensemble de variable et de données correspondant au corps de l'item et à ses métadonnées,
- Les méthodes : ils servent pour communiquer avec des programmes. Ces méthodes contiennent également des règles de décomposition de l'item et de composition de l'item avec d'autres items.

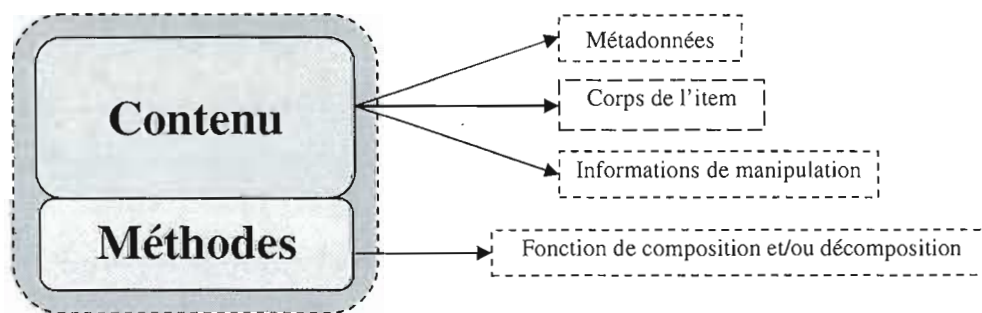


Figure 8.1. Structure d'un objet d'évaluation (OE).

La création d'un OE se fait en utilisant une déclaration comme la suivante :

```
AssessmentObject ao = new AssessmentObject(question1.xml);
```

Les paragraphes subséquents permettront une description des composants d'un OE.

Les métadonnées

Il s'agit de l'ensemble de métadonnées utilisées pour la description de l'item. L'accès aux métadonnées se fait en utilisant la fonction *getMetaData*. Par exemple :

```
ao.getMetaData()
```

Cet appel retourne une structure de données contenant la nature et la valeur de l'ensemble des métadonnées utilisées pour la description de **question1.xml**. Cette structure de données contient des éléments issus de IMS, Dublin Core et de LOM. Certains de ses éléments sont présentés dans la liste ci-dessous. Ceux dont les noms sont précédés de ' * ' sont obligatoires.

- **Identifiant***: Il s'agit du nom du fichier source de l'item. Il est spécifié par l'auteur à la création de l'item ou peut être généré par le système.
- **Titre*** : Il est généralement assez court et descriptif du contenu de l'item.
- **Adaptatif**: spécifie si l'item est adaptatif ou à des paramètres d'item.
- **Langue**: désigne la langue dans laquelle l'item a été créé. Par défaut, la langue est considérée comme français canadien (*fr-CA*).
- **Temporelle**: permet de spécifier si la durée de présentation de l'item au répondant sera comptabilisée et utilisée comme temps de réponse.
- **Auteur(s)***: il contient le(s) nom(s) de(s) personnes qui ont créé(e)s l'item.
- **Mots/Verbes clés***: les mots clés ayant servi à la description de item.
- **Objectifs**: objectif d'évaluation visé par le concepteur de l'item.
- **Taxonomie**: la taxonomie associée à l'item ainsi que le niveau correspondant.

L'interface de saisie des items est présentée à la figure 8.2. Sur cet écran, le premier onglet est sélectionné et présente des champs pour la saisie des métadonnées pour un nouvel item. Certains champs de saisie des propriétés peuvent être masqués ou désactivés dépendamment des choix de l'utilisateur.

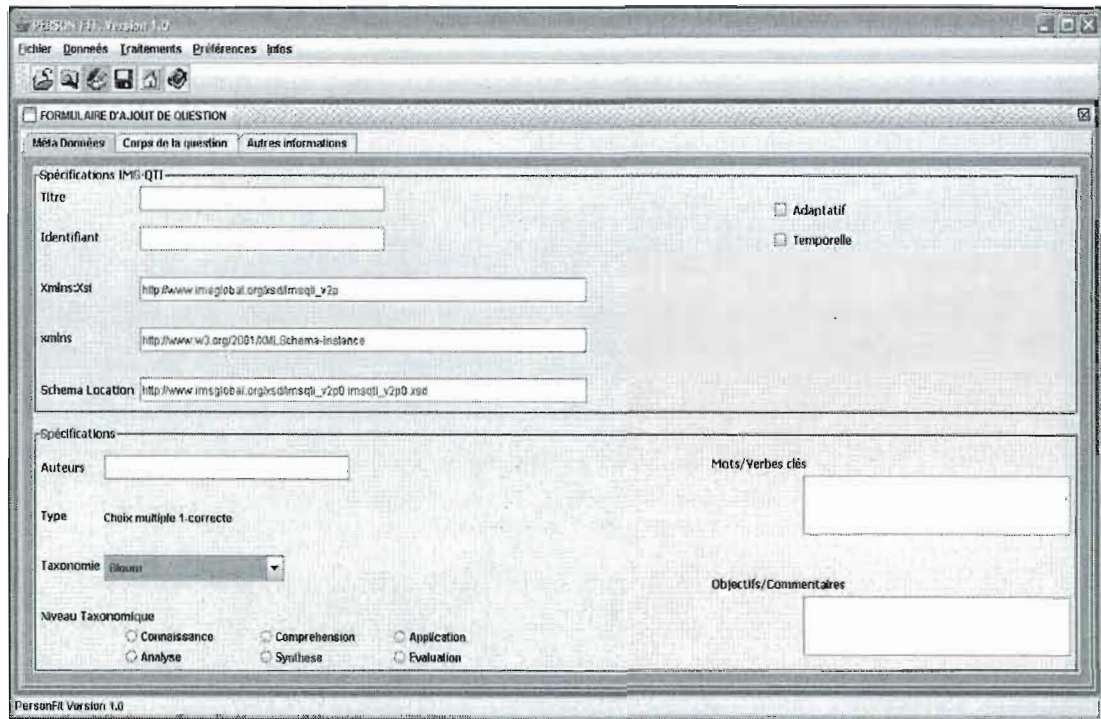


Figure 8.2. Écran d'ajout de métadonnées à un nouvel item

Le corps de l'item

L'accès au corps de l'item associé à l'OE *ao* précédemment défini se fait en utilisant la syntaxe ci-dessous :

```
ItemBody myItemBody = ao.getItemBody()
```

La structure de données retournée est spécifique à chaque type d'item. Il est possible d'accéder directement au type d'un item en utilisant la méthode *getItemType*. Ceci retourne une constante correspondant au type de la question. Les types de questions disponibles actuellement dans PersonFit sont : choix multiples, réponses multiples, réponses ouvertes et vrais/faux. D'autres types d'items sont en cours de

développement. Le contenu du corps de l'item est stocké en utilisant la spécification IMS-QTI.

La figure 8.3 présente l'écran qui permet d'ajouter des données au corps d'une question à choix multiples. Le champ 'Text Rep' de l'écran d'ajout des réponses permet de saisir le texte des choix de réponse. Il est également possible d'ajouter des crédits partiels aux choix de réponses.

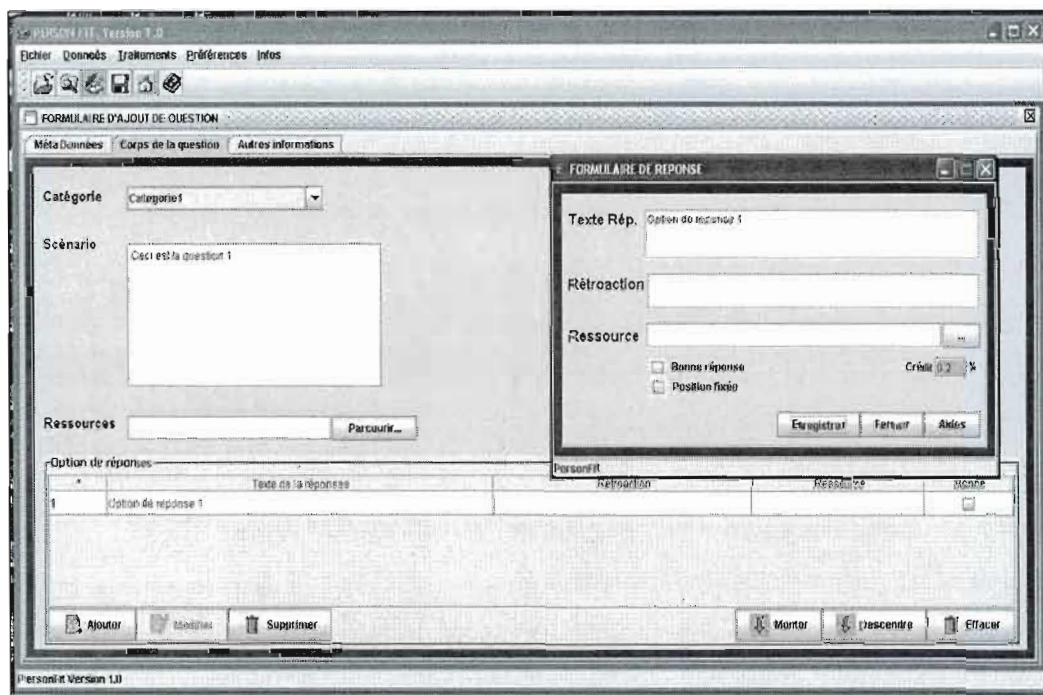


Figure 8.3. Écran d'ajout d'un item à choix multiple

Les informations de manipulations

Ces informations sont utilisées par l'OE dans différentes opérations sur l'item en réponse aux actions des apprenants. Elles sont accessibles en cliquant sur le troisième onglet de l'écran de la figure 8.3. Comme informations de manipulations, il y a le traitement des réponses, le format de présentation, la stratégie d'interaction et les options de partage. Celles-ci seront présentées dans les paragraphes suivants.

Le **traitement des réponses** désigne la façon dont les réponses des apprenants à l'item seront traitées. Certains gabarits sont fournis par défaut. Par exemple pour le gabarit *match_correct* les résultats sont binaires : succès ou échec. Il y a succès si la réponse fournie par le répondant correspond à celle spécifiée comme bonne au moment de la conception de l'item; dans le cas contraire, il s'agit d'un échec. Un autre gabarit est le *map_réponse* qui offre la possibilité de donner des crédits partiels. PersonFit ne dispose pas de moteur de traitement de langage naturel donc les items à réponse ouverte utilise automatiquement le gabarit : *match_correct*.

Le format de présentation désigne le format d'affichage de l'item aux répondants. Le même item peut être présenté différemment selon le profil du répondant en se basant sur la catégorie '*Pédagogie*' de LOM. Des fichiers externes de présentation en différents formats (Flash, Applet java, Java 2D) peuvent être associés aux items.

Les **stratégies d'interaction** désignent la façon et le moment pour envoyer des rétroactions au répondant. Ces rétroactions peuvent être envoyées à l'apprenant immédiatement après avoir validé sa réponse et sont souvent d'ordre correctif. Elles peuvent aussi être envoyées à la fin du test et porter sur certains les items du test ou sur le test au complet. Finalement, les rétroactions peuvent être générées par PersonFit au moment opportun et présenter à l'apprenant. Cette dernière fonctionnalité n'est pas totalement implémentée.

Les **options de partage** concernent trois groupes d'utilisateurs. Le créateur de l'item, les membres de sa *communauté de confiance* et les autres utilisateurs. Le tableau 8.3 représente les droits de partage avec les valeurs numériques associées. Par exemple, pour un item dont le droit le partage est à 731, cela signifie que le créateur de l'item a tous les droits sur l'item, les membres de son groupe peuvent le voir et l'utiliser alors que les autres utilisateurs peuvent seulement le voir. L'absence d'un droit est représentée par le chiffre 0.

	Voir	Utiliser	Modifier	
Créateur	1	2	4	7
Groupe	1	2	4	7
Autres	1	2	4	7

Tableau 8.3. Option de partage des items et valeurs associées

8.2.3 L'interface de présentation des items

La figure 8.4 est une capture de l'interface de présentation des items dans PersonFit. Les boutons de navigation sont désactivés, car cet item est présenté dans un test adaptatif. C'est le tuteur dont une image est présentée dans la fenêtre de droite qui dirige la séquence des questions. Il est tout de même possible de masquer cette fenêtre de droite. Dans cet interface, l'apprenant peut seulement valider son choix ou demander de l'aide. Il est conçu pour présenter les items un à un. En cliquant sur le bouton **Valider** les opérations de validations et de sélection du prochain item sont exécutées. Ensuite, l'item suivant sélectionné est présenté à l'apprenant. En mode non adaptatif, les autres boutons sont activés.

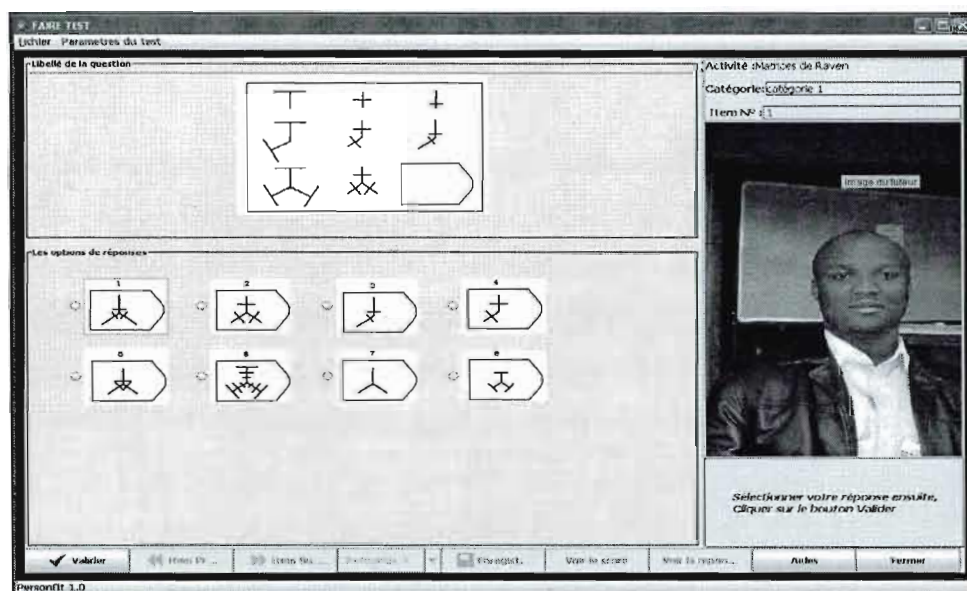


Figure 8.4. Interface client de présentation des items dans PersonFit

En résumé, la section 8.2 nous a permis de présenter la structure et le format des items. Quelques interfaces intervenant dans la conception et l’affichage des items ont été présentées. La prochaine section portera sur l’organisation de la banque d’items au niveau du poste client et du serveur.

8.3 Organisation de la banque d’items

La banque d’items renferme tous les items qui peuvent être utilisés par la plate-forme PersonFit. Deux banques d’items sont disponibles. La première banque est locale et est située sur le poste client de l’utilisateur de PersonFit. Les items de cette banque locale peuvent être exportés dans une seconde banque située sur un serveur. Moodle utilise les items de la banque sur le serveur. Les deux banques d’items sont physiquement constituées par un dossier racine protégé. Dans l’application cliente, le dossier de la banque d’items se trouve dans le répertoire : *personfit/Data*. Sur le serveur du GDAC, la banque d’items se trouve dans le répertoire *moodledata/itemsBank*. À l’intérieur du dossier protégé de la banque d’items se trouvent d’autres dossiers correspondants aux différentes catégories. Les items sont stockés dans ses dossiers. Il y a une organisation hiérarchique des catégories. Ainsi, une catégorie peut avoir une ou plusieurs sous-catégories. En plus de cette structure physique de la banque d’items, une structure logique a été définie. La structure logique est constituée d’un fichier de description au format XML utilisé pour décrire le contenu des catégories. Comme exemple de structure logique nous avons:

- Le fichier placé dans le dossier racine de la banque. Il décrit toutes les catégories ainsi que les relations (hiérarchiques ou participatives) entre elles.
- Un fichier de description est associé à chaque catégorie. Il décrit la catégorie, les items qu’elle contient, la fréquence d’utilisation de chaque item, etc.

- Un fichier manifeste représentant les tests. Chaque test est constitué par un certain nombre d'items sélectionnés ou recommandés par PersonFit.

La figure 8.5 présente un écran de PersonFit montrant la banque locale d'item. Sur la gauche, les catégories sont présentées sous forme de dossier dans l'arborescence. Si le noeud de la catégorie est ouvert, à l'intérieur, il sera affiché l'identifiant des items et le nom des sous-catégories qui s'y trouvent. Le libellé des items de la catégorie sélectionnée est présenté dans la fenêtre de droite. L'onglet '**Question à modélisation**' permet de voir les paramètres des items de la catégorie sélectionnée. Il est également possible de créer, modifier, visualiser ou supprimer des items directement à partir de cet écran.

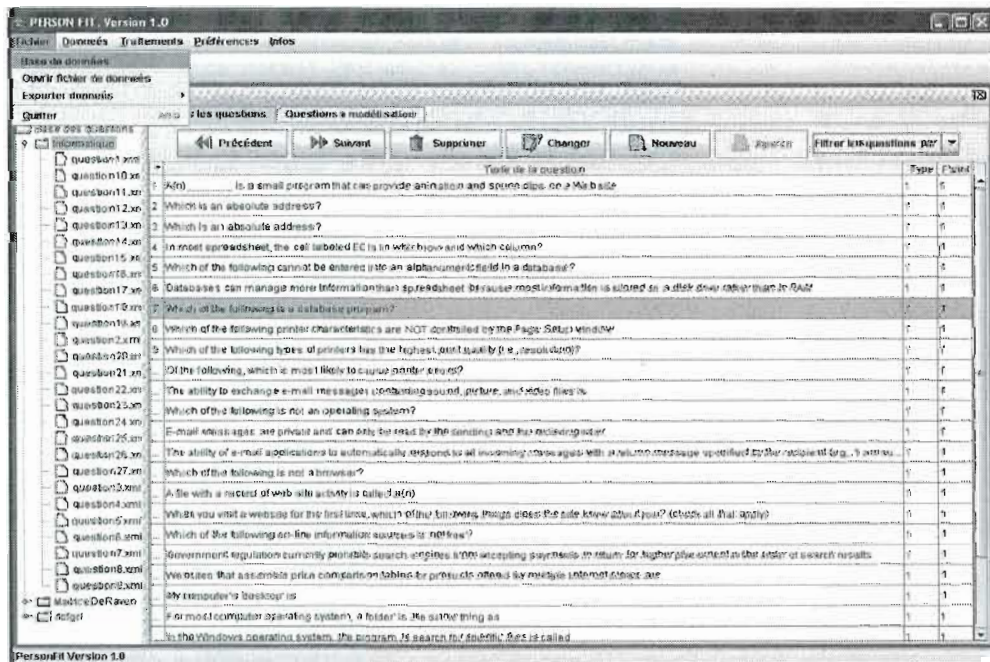


Figure 8.5. Interface de présentation du contenu de la banque locale d'items

La figure 8.6, quant à elle, est une illustration de la banque d'items sur le serveur affichée dans Moodle.

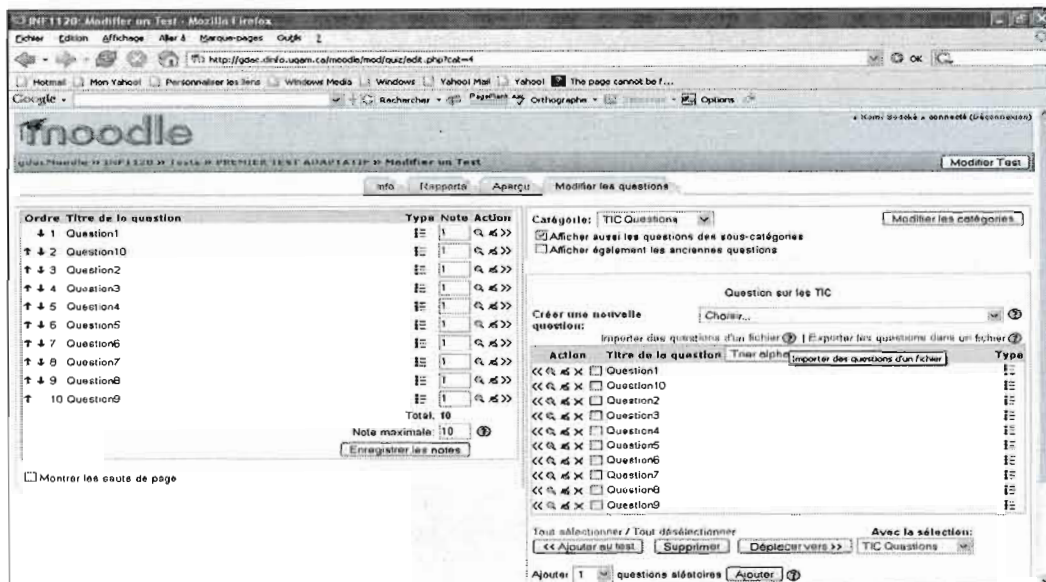


Figure 8.6. Interface Moodle d’affichage de la banque d’items sur le serveur

L’onglet **Modifier les questions** de la figure 8.6 est sélectionné. Il permet de sélectionner les catégories et d’inspecter les items de la catégorie. Ces items peuvent être ajoutés à un test comme c’est le cas dans la fenêtre de droite. Le lien **Importer des questions d’un fichier** permet d’importer les questions de la banque locale d’items vers la banque d’items sur le serveur. La banque d’items à importer doit être comprimée en utilisant un logiciel comme *WinZip*. Des programmes PHP qui s’exécutent sur le serveur se chargent de décompresser le fichier, de vérifier l’état et la structure de tous les fichiers sources des questions ainsi que des images et autres ressources associées. Le bilan de cette analyse est présenté à l’écran. Après validation de la part de la personne qui soumet les items, ces derniers sont ajoutés à la banque d’item sur le serveur dans les catégories concernées.

Le lien **Exporter des questions dans un fichier** permet d’exporter les questions de la banque d’items sur le serveur vers la banque locale d’items. Ces items peuvent

également être téléchargés par les personnes autorisées en tenant compte des droits associés.

En résumé, les deux banques d'items ainsi que les différents composants de PersonFit et de Moodle, interagissent dans le processus de conception et d'administration des tests adaptatifs. Le prochain chapitre présentera une perspective intégratrice de ses différents composants avec un schéma explicatif à l'appui.

8.4 Intégration et interactions avec Moodle

La figure 8.7 présente les principaux composants qui interviennent dans l'intégration et l'interaction de PersonFit avec Moodle. Nous pouvons citer, entre autres :

- **L'application cliente de PersonFit** qui sert de pont entre la banque locale d'items et la banque d'items sur le serveur. Elle permet également de créer, de modifier et de supprimer des items dans la banque locale d'item. Finalement, elle peut servir à récupérer le profil des répondants, à calibrer les items localement, etc.
- Le **ThetaEstimator** : Il sert principalement à l'estimation du niveau d'habileté (θ), de sa précision (S_θ) et d'un indice d'ajustement du patron de réponses (L_z). Il est écrit en langage C et est appelé par les codes PHP de Moodle.
- Le **ItemFinder** : Ce composant se charge principalement de la recherche du prochain item à administrer au répondant. Il récupère les valeurs calculées par le ThetaEstimator, consulte la table d'information et récupère dans la banque d'items sur le serveur un groupe contenant trois items optimaux. Ensuite, l'item à administrer est sélectionné de façon aléatoire parmi ce groupe d'items. Ceci permet de minimiser les chances que des apprenants, ayant le même niveau d'habileté (θ), reçoivent tout le temps le même item.

- Le **ItemPresenter**, il s'agit du composant servant pour l'affichage des items au répondant. Comme PersonFit est intégré dans Moodle, le ItemPresenter ici est une modification de l'interface web d'affichage des questions de Moodle.

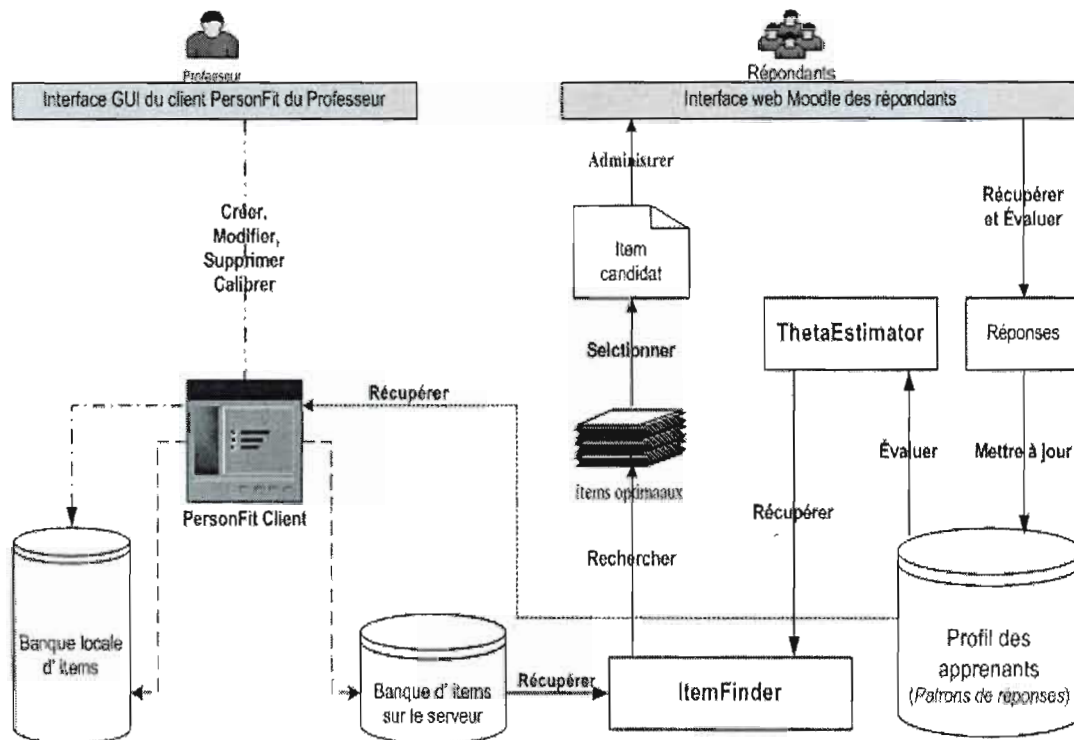


Figure 8.7. Architecture d'interaction entre les composants de PersonFit et Moodle

Pour illustrer l'interaction de ces différents composants lors d'un test adaptatif, un extrait d'un déroulement normal et simplifié de test adaptatif dans PersonFit sera présenté dans le paragraphe suivant.

Cet extrait fait intervenir les différents composants présentés à la figure 8.7 ainsi que le concept de table d'information. Pour cet extrait, nous avons tenu compte des considérations suivantes:

- Le nom de l'apprenant en situation de test est : *Marc*.
- La règle d'arrêt est une précision de 0,2 sur la valeur estimée du niveau d'habileté (θ) ou un maximum de 20 items admirés.
- La table d'information du tableau 8.4 sera utilisée. Dans la parenthèse à droite de chaque item est indiquée la catégorie à laquelle ils appartiennent.

θ	Item par ordre de préférence de la gauche vers la droite pour chaque valeur de θ				
-1	125(C1)	18(C2)	52(C1)	25(C1)	78(C2)
0	2(C2)	24(C1)	18(C1)	56(C2)	8(C2)
1	5(C1)	25(C2)	85(C2)	19(C1)	12(C1)

Tableau 8.4. Exemple de table d'information pour simulation

Scénario d'utilisation

1. Marc se connecte à Moodle en utilisant son nom d'utilisateur et son mot de passe. Il sélectionne son cours de base de données en ligne. En cliquant sur le chapitre des '*traitements de requêtes*', il remarque qu'un test est actif pour lui et décide de le faire.

2. Comme Marc sélectionne le test, PersonFit procède à l'initialisation de son modèle d'apprenant et estime son niveau d'habileté θ_{initial} à -1 pour le concept concerné. Cette valeur est arbitraire et est choisie uniquement pour cet exemple. Une structure de données représentant la table d'informations est chargée dans la mémoire de l'ordinateur. Toutes les données d'initialisation sont chargées dans l'état des connaissances à court terme de Marc.

3. Pour $\theta = -1$, le **ItemFinder** sélectionne les trois items les plus informatifs qui sont les items 125, 18 et 52 dans le tableau 8.4. Ensuite, l'item 18 est sélectionné de façon aléatoire parmi ce groupe d'item. Cet item est ensuite retiré de la table d'informations.

4. Le **ItemPresentator** (ici l'interface web de Moodle pour les apprenants) présente l'item 18 à Marc et attend sa réponse.

5. Marc donne sa réponse à l'item 18.

6. Le **ThetaEstimator** calcule la nouvelle valeur de son niveau d'habileté. Les valeurs suivantes sont obtenues :

- Le niveau d'habileté : $\theta = -0,2$.
- L'erreur type sur la valeur estimée du niveau d'habileté : $S_{\theta} = 0,35$.
- L'indice d'ajustement du patron de réponse : $L_z = 0,4$.

7. Comme la règle d'arrêt n'est pas satisfaite (la valeur de $S_{\theta} < 0,2$), le **ItemFinder**, récupère les valeurs estimées par le **ThetaEstimator** et recherche ensuite dans la banque d'items un groupe d'items optimaux qui sera constitué des items 2, 24 et 56. Le choix de ces items s'explique par le fait qu'ils sont les items les plus informatifs non administrés à l'apprenant pour le niveau d'habileté 0. Dans la table d'information, cette valeur 0 est celle la plus proche de la valeur courante du niveau d'habileté qui est de -0,2.

8. Deux autres contraintes sont également prises en compte dans la sélection du prochain item : la couverture de contenu et la fréquence d'exposition des items.

- Les *contraintes de couverture de contenu* peuvent être intrinsèques ou spécifiées par l'auteur du test. Supposons que les items 18(C1) et 56(C2) sont deux items optimaux potentiels, mais que plusieurs items de la catégorie C1 ont été déjà administrés, alors l'item 56(C2) sera privilégié. Les contraintes

intrinsèques résultent le plus souvent de relation de pré-requis entre les différents concepts sur lesquels portent l'évaluation. Par exemple, il sera inutile d'administrer un item portant sur la division à Marc s'il a déjà échoué des items portant sur l'addition et la soustraction, sauf si nous voulons détecter s'il est en train de jouer avec le système (*gaming*).

- Les *contraintes de la fréquence d'exposition des items*; elles consistent à privilégier un item par rapport à un autre sur la base du nombre de fois que ce dernier a été administré aux apprenants. Donc, de plusieurs items optimaux potentiels, celui qui a le plus faible taux d'administration aux apprenants est privilégié.

Le **Profiler** est un processus qui est lancé au démarrage du test et qui tourne en arrière-plan pour faire une analyse de la séquence des items. Il intervient dans le processus de sélection d'un prochain item et est prioritaire sur les deux contraintes précédentes. Le **Profiler** tente de bâtir un profil pour Marc en se basant sur la séquence des items administrés ainsi que les réponses obtenues. Il peut donc recommander des items au **ItemFinder** afin d'établir certaines évidences concernant les connaissances de **Marc** dans les concepts concernés par le test. Par exemple après avoir administré 12 items à Marc, le **Profiler** remarque que 7 questions portent sur le concept C1 et 5 sur le concept C2. *Marc* a trouvé la bonne réponse aux 5 items portant sur C2 mais a seulement trouvé deux bonnes réponses sur les sept items du concept C1 qui lui ont été administrés. Le **Profiler** peut commencer par établir l'évidence de la connaissance du concept C2 et d'une méconnaissance du concept C1. Pour confirmer cette hypothèse, certains items supplémentaires de connaissance du concept C1 seront recommandés par le **Profiler** au **ItemFinder** en tenant compte de leur information fournie.

9. Les étapes 3 à 8 sont exécutées tant que S_0 est inférieur à 0,1 ou lorsque moins de 20 items sont administrés. Supposons qu'après 14 items une valeur de S_0 égale à 0,1

est obtenue; un message généré par PersonFit est envoyé à Marc. Ce message le remercie d'avoir complété le test et lui affiche son score si besoin est. Son profil (principalement état des connaissances à long terme et la mémoire épisodique) est mis à jour dans la banque des profils sur le serveur.

8.5 Conclusion du chapitre

Ce chapitre a principalement porté sur l'implémentation de PersonFit. Pour ce faire, les choix technologiques effectués pour le développement des différents composants ont été présentés. Également, une présentation du fonctionnement de ces composants avec quelques captures d'écran à l'appui a été faite suivie d'un scénario d'utilisation faisant intervenir les composants. Après l'implémentation de PersonFit et l'intégration dans Moodle, des tests pratiques et des simulations ont été effectués afin de vérifier son efficacité. Le prochain chapitre portera sur les résultats obtenus. Une discussion sera également effectuée.

CHAPITRE IX

LES RÉSULTATS OBTENUS

Ce chapitre permettra de présenter les résultats et les discussions sur le travail effectué. Les résultats obtenus pour les différentes fonctionnalités implémentées seront présentés dans cette section. Il s'agit principalement de l'algorithme de tests adaptatif, de la détection de jeu sur la base des patrons de réponses inappropriés et du diagnostic cognitif de l'apprenant.

9.1 Résultat de l'utilisation des tests adaptatifs

Plusieurs tests pratiques et des simulations ont été effectués en vue de déterminer l'efficacité de l'algorithme de test adaptatif implémenté dans PersonFit. Dans les expérimentations, les données et les paramètres d'items provenant du TCALS (*Test de Classement en Anglais Langue Seconde*) du Québec fournies par le laboratoire CAMRI ont été utilisés. La version non adaptative du TCALS compte 85 items administrés à tous les répondants. C'est un test administré à grande échelle, ce qui permet d'avoir des milliers de patrons réponses provenant d'une situation réelle d'évaluation. L'objectif des expérimentations est de simuler une version adaptative

du TCALS en utilisant un échantillon des fichiers de données du TCALS. Nous tenterons de répondre à trois interrogations.

Première interrogation: Quel est le nombre d'items nécessaire pour atteindre la convergence de la valeur du niveau d'habileté des apprenants de l'échantillon sélectionné?

Deuxième interrogation : Quel est l'impact d'un mauvais choix de la valeur initiale du niveau d'habileté? Cette interrogation se justifie par le fait qu'aucune information n'est disponible sur les apprenants de l'échantillon pour servir à l'initialisation de leurs modèles.

Troisième interrogation : Est-ce qu'il est possible de minimiser, voire annuler le biais engendré par ce mauvais choix de la valeur initiale du niveau d'habileté? Si tel est le cas, quel est le nombre d'items nécessaires pour le faire?

Pour les expérimentations, la première valeur du θ_{initial} est fixée à -3. Chaque item est sélectionné en utilisant la procédure de sélection de l'algorithme de test adaptatif adoptée (section 6.2) et est administré à un échantillon de 515 répondants. À chaque item administré, la moyenne de la valeur estimée de θ_i grâce à la méthode AMAP est calculée. Cette moyenne est représentée; ceci permet d'obtenir la courbe en pointillée débutant à -3 sur la figure 9.1. La même opération est répétée avec une valeur de θ_{initial} de -2, ainsi de suite jusqu'à 3. La figure 9.1 représente le résultat de l'expérimentation d'une version adaptative du TCALS pour l'échantillon de 515 répondants en prenant différentes valeurs pour le θ_{initial} .

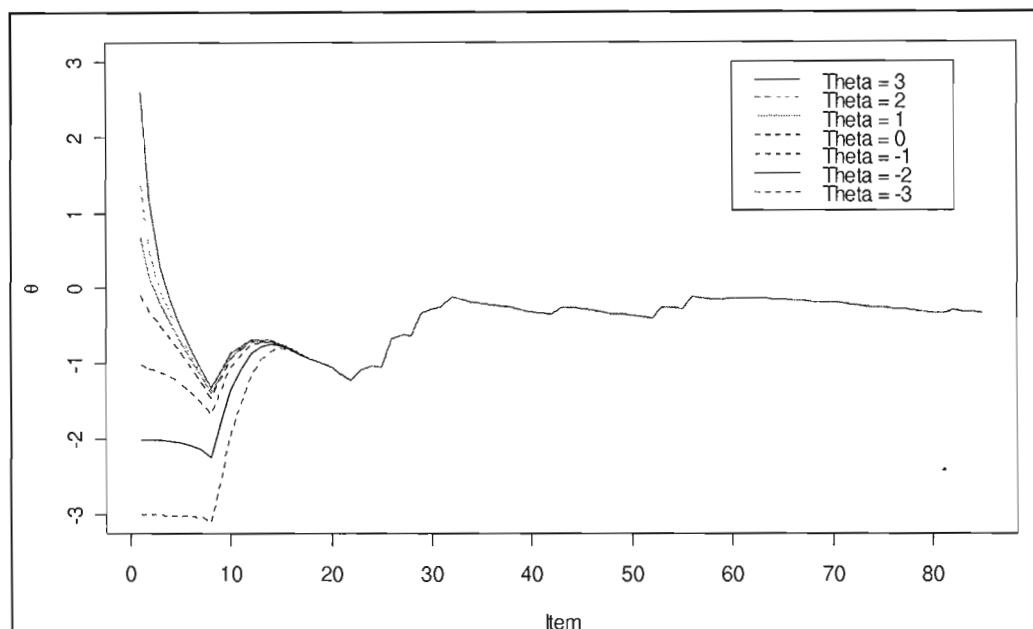


Figure 9.1. Évolution du calcul du niveau d'habileté pour les valeurs de θ_i

À partir de cette figure, il y a les observations suivantes à faire :

Observation 1 : Au delà de 15 items, toutes les courbes se confondent quelque soit la valeur du θ_{initial} .

Interprétation 1: La deuxième interrogation portant sur l'impact du mauvais choix de la valeur initiale du niveau d'habileté des apprenants de l'échantillon trouve sa réponse. Avec moins de 15 items, le θ_{initial} a un impact sur la valeur calculée de θ , mais au-delà du 15^{ème} item celle-ci est indépendante du θ_{initial} . La troisième interrogation trouve également sa réponse, car la convergence de toutes les courbes indique qu'il est possible d'annuler l'effet de la valeur du θ_{initial} . En somme, l'**observation 1** justifie le choix d'une bonne stratégie d'initialisation du modèle de l'apprenant surtout si le test pourrait être constitué de peu d'items.

Observation 2: Au-delà de 55 items, il n'y a plus de variation significative de l'allure de la courbe.

Interprétation 2 : Cette observation montre qu'il n'y a plus de variation significative de l'estimateur du niveau d'habileté. L'ajout d'un item supplémentaire n'apporte pas plus d'informations. Ceci indique que la convergence est obtenue. La première interrogation trouve ici sa réponse; il a fallu 55 items pour atteindre la convergence de la valeur du niveau d'habileté des apprenants de l'échantillon sélectionné. Par rapport à la version non adaptative qui compte 85 items, la version adaptative du TCALS a nécessité 35% moins d'items pour estimer le niveau d'habileté avec la même précision pour tous les répondants.

Plusieurs autres simulations de répondants à des items fictifs dont les paramètres sont connus ont été effectuées avec le logiciel statistique R¹⁴. L'objectif visé est de comparer les résultats en version adaptative et classique de ces tests. En règle générale, nous arrivons à obtenir la valeur du niveau d'habileté, et avec plus de précision, en utilisant environ 35 % moins d'items pour les versions adaptatives de ces tests.

9.2 Résultat de la détection de jeu

La détection de jeu dans PersonFit est réalisée principalement sur la base des patrons de réponses inappropriés. Des simulations d'apprenants joueurs ont été effectuées pour vérifier l'efficacité de la stratégie de détection proposée.

Pour utiliser la simulation comme mécanisme pour vérifier les résultats potentiels à obtenir, différentes contraintes doivent être mis sur les modèles simulés pour les rendre le plus près que possible de la réalité. Dans le cas qui nous concerne, pour la simulation des apprenants, les contraintes suivantes seront considérées:

¹⁴ <http://cran.r-project.org>

- Il peut arriver que deux ou plusieurs répondants, ayant le même niveau d'habileté, aient des patrons de réponses plus ou moins différents.
- Les stratégies utilisées par les apprenants qui tentent de jouer peuvent être différentes d'un apprenant à un autre. Pour identifier ces stratégies, les résultats de recherches antérieures menées par Raïche et Blais (2002) portant sur l'identification et le dépistage des apprenants qui tentent de se sous-classer ont été considérés. Deux stratégies ont été retenues pour être utilisées comme contrainte lors des simulations. La première contrainte consiste à supposer que, pour jouer dans la plate-forme, certains apprenants peuvent répondre au hasard aux items qui leurs sont administrés. La deuxième contrainte porte sur les apprenants qui vont donner la mauvaise réponse aux items au cas où ces derniers supposeraient qu'ils connaissent la bonne réponse. Ces items reçoivent donc des réponses inversées.
- Par rapport aux résultats obtenus dans le TCALS, les apprenants sont classés dans quatre niveaux de cours d'anglais.

En tenant compte des contraintes précédentes, Raïche et Blais (2002) ont procédé à la simulation par ordinateur de 1000 patrons de réponses selon quatre niveaux d'habileté entre -3 et 3 et selon cinq conditions :

1. des patrons de réponses totalement normaux;
2. des patrons de réponses où 10 % des items reçoivent des réponses produites au hasard;
3. des patrons de réponses où 20 % des items reçoivent des réponses produites au hasard;
4. des patrons de réponses où 10 % des items reçoivent des réponses inversées;
5. des patrons de réponses où 20 % des items reçoivent des réponses inversées.

Les figures 9.2 et 9.3 présentent les résultats de la détection obtenus en utilisant l'indice L_z . Les taux de détection des patrons de réponses inappropriés en fonction

du niveau d'habileté réel des apprenants sont représentés par les histogrammes sur ces figures.

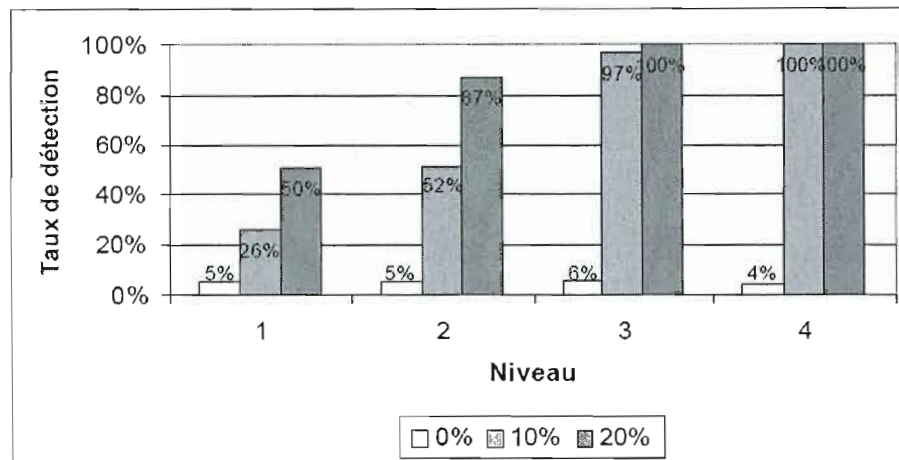


Figure 9.2. Résultat de la détection lorsque les réponses sont données au hasard

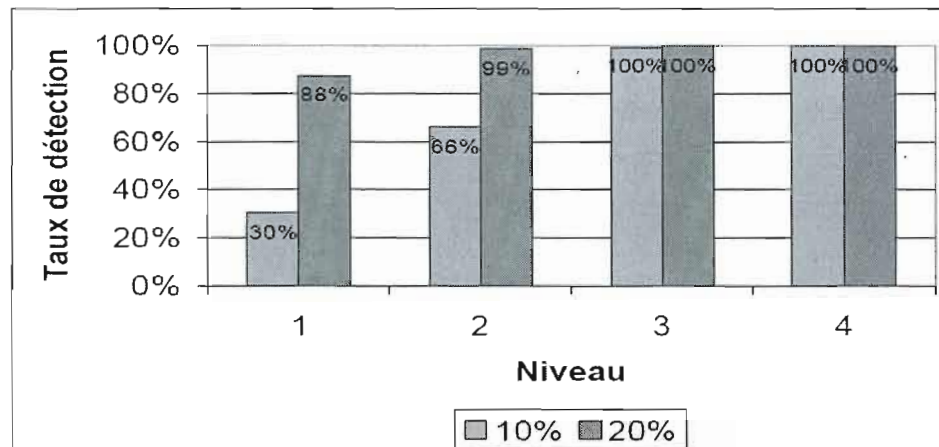


Figure 9.3. Résultat de la détection lorsque des réponses incorrectes sont données

Les figures 9.1 et 9.2 indiquent que les taux de détection sont de plus de 90% pour les niveaux 3 et 4 avec 10 à 20% de réponses inappropriées. Ces résultats attestent de l'efficacité de l'indice Lz en matière de détection des patrons de réponses inappropriées pour les apprenants de hauts niveaux d'habileté qui tentent de jouer.

Toutefois, même si le jeu est supposé comme la source principale des patrons de réponses inappropriées détectés, il importe d'apporter quelques nuances. En effet, ces patrons peuvent également provenir d'autres sources comme, par exemple une inattention, un stress, etc. de la part de l'apprenant. De ce fait, les patrons de réponses détectés doivent être récupérés par un tuteur humain en utilisant PersonFit. Ceux-ci sont ensuite examinés pour vérifier la pertinence de la détection. Il revient donc à ce dernier la décision à prendre quant aux mesures correctives. La figure 9.4 présente le résultat de la détection par PersonFit sur un fichier de données de 523 élèves. Le tuteur ou l'enseignant peut exporter ces données en fichier CSV ou pdf, en valider le contenu et prendre les mesures correctives adéquates.

PERSON FIT, Version 1.0

Fichier Données Traitements Préférences Infos

FORMULAIRE DES FICHIERS DE DONNEES

Options de détection

CONTENU DU FICHIER DE DONNEES : 523 Elèves

Lancer la détection

RESULTAT DE LA DETECTION

Exporter le résultat vers

RESULTAT DE LA DETECTION : 52 Etudiants

Modifier la détection

	MATRICULE	COTE	NOTE	MAP	SMAP	LZ
1	0570963	...	54	-0,31	0,16	-2,67
2	053	...	79	0,65	0,23	-1,99
3	0567050	...	79	0,56	0,22	1,79
4	0560802	...	78	0,74	0,25	-1,16
5	0530141	...	64	-0,25	0,16	-2,19
6	0543999	...	55	-0,67	0,16	-3,22
7	0578993	...	45	-1,59	0,20	-2,80
8	0561582	...	78	0,61	0,23	-1,92
9	0563705	...	67	-0,24	0,16	-1,79
10	0537814	...	67	-0,07	0,17	-1,61
11	0503730	...	70	-0,00	0,17	-1,60
12	0556844	...	83	1,38	0,39	-0,17
13	0583722	...	75	0,29	0,19	-1,25
14	0505051	...	77	0,56	0,22	-1,26
15	0538209	...	68	-0,15	0,16	-1,88
16	0500314	...	79	0,50	0,23	-1,88
17	0595090	...	81	1,08	0,31	-0,94
18	0543293	...	83	1,29	0,36	-0,44
19	0532564	...	72	0,16	0,18	-1,19
20	0587820	...	67	-0,14	0,16	-1,52

PersonFit 1.0

Figure 9.4. Présentation par PersonFit du résultat de la détection

9.3 Résultat du diagnostic cognitif des apprenants

Des résultats concrets n'ont pas pu être obtenus en ce qui concerne la mise en application des stratégies de diagnostic cognitif proposées. Des modifications et des ajouts de codes ont été effectués dans la version de Moodle installée sur le serveur du GDAC afin d'y intégrer PersonFit. L'adresse pour y accéder est la suivante: <http://camri.uqam.ca/admin/moodleDev/>

Ensuite, tous les chapitres et sections du cours de programmation I en Java donné à l'UQAM ont été ajoutés de même que quelques items d'évaluations organisés autour des concepts clés enseignés dans le cours. L'objectif visé est de mettre la plate-forme à la disposition des professeurs qui donnent le cours en question. Malheureusement, la plate-forme n'a pas été utilisée; donc les profils d'apprenants (principalement état des connaissances à long terme et mémoire épisodique) sur lequel PersonFit pourrait se baser pour effectuer des diagnostics n'ont pu être obtenus. Néanmoins, le site est gardé actif et des démarches seront effectuées durant la session d'automne 2007 en vue de voir dans quelle mesure il pourrait être utilisé.

9.4 Conclusion du chapitre

Somme toute, les résultats d'implémentation obtenus aussi bien sur des données réelles que par des simulations, ont été concluants. Ils indiquent la pertinence du travail effectué et l'efficacité des algorithmes et des stratégies proposées. Dans la prochaine section, qui est la conclusion du mémoire, en nous basant sur les résultats obtenus nous allons dégager nos principales contributions. Également, les améliorations à apporter au travail effectué ainsi que les perspectives seront présentées.

CHAPITRE X

CONCLUSION

La conclusion de ce mémoire nous permettra de faire une récapitulation des principales réalisations et contributions. Cette synthèse permettra de ressortir nos contributions ainsi que les futurs axes d'améliorations ou d'explorations. Également, nous présenterons les perspectives de recherches et les améliorations pouvant être apportées en vue de faciliter l'utilisation de PersonFit.

10.1 Synthèse

Dans le cadre de ce mémoire, nous avons travaillé sur la conception et l'implémentation d'une plate-forme d'évaluation adaptative des apprentissages et de diagnostic cognitif des apprenants. Notre démarche a consisté à:

- Identifier les limites des tests classiques; ce qui a mené à des propositions de solutions en utilisant les tests adaptatifs.
- Faire un état de l'art sur l'adaptation; ce qui nous a permis de constater que peu de travaux ont été effectués en matière d'adaptation des tests en ligne.
- Identifier les exigences qui rendent difficile la conception de plate-forme d'administration des tests adaptatifs par ordinateur.
- Rechercher des modèles de calcul et de représentation pouvant répondre à ces exigences. La TRI et les RB ont été retenus comme modélisations.

- Proposer une architecture d'implémentation basée sur celle des STI.
- Concevoir et implémenter la plate-forme *PersonFit* dans sa version 1.0
- Ajouter deux fonctionnalités importantes à *PersonFit*, notamment la détection de jeu chez les apprenants et le diagnostic cognitif des apprenants.
- Présenter les résultats de l'utilisation de *PersonFit* et des principes qui soutenant les fonctionnalités y ont été implémentées.

En résumé, l'utilisation adéquate des RB et de la TRI nous a donc permis de mettre en oeuvre une plate-forme fonctionnelle d'administration d'évaluation adaptative des apprentissages. Elle a ensuite été intégrée dans la plate-forme de E-learning Moodle; ce qui nous a permis d'étendre les fonctionnalités de Moodle au partage de différentes banques d'items au format IMS-QTI 2 et l'administration d'évaluations adaptatives aux apprenants. Les résultats de ces évaluations, après un processus de détection pour s'assurer que l'apprenant n'est pas en train de jouer avec la plate-forme, sont utilisés pour un diagnostic cognitif des apprenants.

Les sujets abordés dans ce mémoire, notamment l'adaptation des évaluations selon le niveau d'habileté cognitive de l'apprenant, la détection du jeu et le diagnostic cognitif, constituent actuellement d'actifs sujets de recherches. La finalité d'un travail effectué dans un contexte de recherche étant d'apporter des contributions dans le domaine concerné, nous allons, dans la prochaine section, présenter nos principales contributions.

10.2 Contributions

Différentes contributions ont été apportées dans les domaines concernés par ce mémoire. Ainsi, dans le domaine des tests adaptatifs, selon le niveau des habiletés cognitives des apprenants, l'approche se veut innovatrice. En effet, comme mentionné

dans l'état de l'art, même si l'adaptation a fait l'objet de nombreuses recherches et implémentations, ces dernières ont principalement portées sur l'apprentissage. Un petit nombre de travaux ont donc été menés dans ce domaine. Comme contribution, nous proposons de combiner les acquis en matière d'adaptation de tests en éducation avec des modélisations informatique. Ceci nous a mené à l'utilisation conjointe de la TRI qui est une modélisation utilisée en psychométrie et des RB issus de l'informatique. Une combinaison des deux modélisations nous a permis de bâtir la plate-forme PersonFit en prenant comme exemple d'architecture celle des STI.

En ce qui concerne la détection du jeu, différentes approches de solutions ont été utilisées. Ces approches sont principalement basées sur des indicateurs provenant d'une analyse du comportement de l'apprenant. Par exemple, des indicateurs comme le nombre d'aides demandés par l'apprenant pendant la résolution d'un problème, le temps de réponse, etc. sont utilisés (Baker et *al.*, 2004). À notre point de vue, ces indicateurs utilisés sont subjectifs et ne garantissent pas de faire des conclusions objectives, car nous estimons qu'un ensemble de comportements observés chez un apprenant ne suffit pas pour tirer des conclusions évidentes de jeu chez ce dernier. Selon nous, une méthode plus objective serait d'ajouter à l'approche actuelle l'analyse de la cohérence des réponses fournies par l'apprenant à chacun des items du test et l'ajustement global de son patron de réponse par rapport à la valeur estimée de son niveau d'habileté. Ceci constitue la contribution principale en ce qui a trait à la détection du jeu. Pour l'analyse de la cohérence des réponses fournies, nous nous basons sur le fait que les conséquences du jeu se traduiront de la part de l'apprenant par des patrons de réponses inappropriés, plutôt étranges et ainsi peu probables. Un indice approprié issu de la TRI qui est appelé indice de l'ajustement du patron de réponse, noté L_z , a été mis à contribution. Cette approche constitue l'originalité de la stratégie de détection de jeu utilisé dans la plate-forme PersonFit.

Finalement, en ce qui concerne l'implémentation du diagnostic cognitif dans PersonFit, elle a été assez délicate. En effet, dans les STI classiques, le diagnostic est souvent basé sur une banque des erreurs les plus courantes et les connaissances reliées à ces erreurs. Cette approche est adaptée au STI car ils sont généralement utilisés comme support pour l'enseignement dans un domaine précis. Étant donné que la plate-forme PersonFit n'est pas conçue pour un domaine précis, la mise en place de la banque des erreurs sera difficile. Le principe du diagnostic utilisé dans PersonFit a donc été basé sur la prescription. Il s'agit de prendre comme modèle celui des STI ensuite d'observer un échec et d'identifier les pré-requis qui pourraient en être la cause. Le RB sous-jacent à l'état des connaissances à long terme de l'apprenant qui est un recouvrement du domaine des connaissances a été mis à contribution. L'algorithme MPE (Most Probable Explanation) est utilisé pour permettre une interprétation des échecs de l'apprenant (Nkambou et Tchétagni, 2004). Cet algorithme existe déjà. La contribution consiste dans la tentative de l'intégrer dans une plate-forme développée.

Les publications acceptées, les conférences auxquelles nous avons participées en vue de présenter le travail effectué ainsi que les résultats obtenus attestent de ces contributions. La liste de ces publications et colloques est présentée en annexe de ce mémoire.

D'autres contributions ont été faites sur le plan pratique. Le choix des langages de programmation notamment Java du côté client et PHP-MySQL du côté serveur qui sont tous des technologies libres facilitera l'accessibilité des codes sources. Les programmes écrits sont disponible au CAMRI et peuvent être récupérés en guise d'exemple par d'autres équipes de recherches ou un étudiant intéressé par le sujet sans contrainte de droit d'auteur ou de plate-forme. Les codes de la version cliente de PersonFit ont été bien documentés en utilisant le format *JavaDocs*, ce qui va simplifier leur lecture via les pages web générées. Ces codes ont été également

regroupés en plusieurs librairies autonomes ce qui va faciliter la maintenance et l'évolution.

Au terme de ce travail, nous affichons une grande satisfaction car la majorité des objectifs fixés ont été atteints et nous avons apporté des contributions à des sujets de recherches qui nous intéressent. Nous disposons également d'éléments d'informations pour améliorer les résultats. Mais dans l'état actuel des développements, toutes les fonctionnalités ne sont pas parfaitement opérationnelles et dans certains cas, il y a nécessité d'aller modifier la valeur de certaines constantes à l'intérieur des codes pour obtenir les résultats attendus. Il y a donc certaines améliorations à apporter pour rendre la plate-forme complètement opérationnelle. Dans la prochaine section ces améliorations seront présentées ainsi que des perspectives de recherche et développement que suscite le travail effectué.

10.3 Améliorations et perspectives

Les résultats obtenus et contributions nous motivent à finaliser le développement de certaines des fonctionnalités de PersonFit. Ces améliorations ajouteront de la fiabilité et de la robustesse à la plate-forme. Au nombre de ces améliorations, il y a quelques corrections et aides à ajouter aux interfaces graphiques pour les rendre plus conviviales et faciles d'utilisation pour des utilisateurs novices en informatique. Il faudrait également ajouter un meilleur contrôle des erreurs avec des messages d'erreur adéquats ce qui pourrait aider également les utilisateurs novices.

Pour un développeur intéressé, mentionnons que les codes sources de la fonctionnalité de diagnostic cognitif ne sont pas totalement terminés avec l'implémentation de l'algorithme MPE. Néanmoins, des exemples de codes sources ainsi que la librairie eBayes ont été fournis en vue de servir et de guider de futures implémentations.

Mis à part ces améliorations à apporter; le travail effectué suggère certaines perspectives de recherches intéressantes qui seront présentées dans les prochains paragraphes.

Nous expérimentons des possibilités de reconnaissance et d'expression d'émotions dans la plate-forme PersonFit. Les résultats de projets antérieurs notamment Émilie 1 et Emilie2 du laboratoire GDAC seront utilisés à cette fin. L'image du tuteur dans la version actuelle du *ItemPresentator* (Figure 8.4.) serait alors remplacée par un tuteur émotif et s'exprimant en langage naturel.

Il existe d'autres modélisations de la TRI qui peuvent être étudiées en vue de rendre la plate-forme plus flexible. Il s'agit des modèles intégrant les crédits partiels aux réponses de l'apprenant. Également, certaines de ces modélisations intègrent des concepts de classes latentes de difficulté d'apprentissage. Ceci pourra être utilisé dans le diagnostic cognitif des apprenants. Finalement, certaines de ces modélisations prennent en considération l'estimation conjointe du niveau d'habileté de l'apprenant et de l'ajustement de son patron de réponses. Les résultats de ces modélisations pourraient être étudiés et comparés avec l'utilisation de l'indice *Lz*.

Le composant *ItemFinder* est en constant développement, car c'est le cœur de la plate-forme PersonFit. Comme perspectives de recherche, différentes heuristiques peuvent être étudiées et ajoutées, ce qui pourrait avoir un impact sur la vitesse de convergence de la valeur du niveau d'habileté. En guise d'exemple, une évidence toute simple pourrait être considérée. Un apprenant à qui un item est administré le réussit ou l'échoue. Une projection pourrait être faite en considérant ces deux éventualités. Ainsi, à une étape *n*, une condition supplémentaire serait ajoutée à la sélection du prochain item et portera sur le fait que cet l'item à sélectionner doit fournir une information maximale en cas de réussite ou d'échec.

Actuellement la banque est centrale et située physiquement sur le serveur du GDAC. Il serait intéressant de considérer l'option d'une banque d'items distribuée à laquelle la plate-forme peut accéder. Une banque d'items donnée peut disposer de métadonnées ou de la table d'information d'autres banques. PersonFit pourrait dans ce cas accéder à un item optimal provenant d'une autre banque d'items en cas de besoin.

Sur la base des résultats de l'apprenant et de son diagnostic cognitif, il serait intéressant d'avoir une fonctionnalité de recommandation de cheminement d'apprentissage à l'intérieur d'un cours mis en ligne dans Moodle. De récentes spécifications de IMS, notamment le IMS *Simple Sequencing* (<http://www.imsglobal.org/simplesequencing/index.html>), seraient alors mises à contribution.

Ces améliorations et perspectives constitueraient des sujets de recherche intéressants pour une suite de nos études au niveau du doctorat.

Bibliographie

(Aïmeur et Hage, 2005) Hage, H. and Aïmeur, E. *Exam Question Recommender System*. Proceedings AIED (*Artificial Intelligence for EDucation*) 2005.

(Appadurai, 1996) Appadurai, A. (1996). *Modernity at Large. Cultural Dimensions of Globalization*. Minneapolis. University of Minnesota Press.

(Anderson, 1983) Anderson, J. (1983). *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.

(Baker et al., 2006) Baker, R. S., Corbett, A.T., Koedinger, K.R., Evenson, S., Roll, I., Wagner, A. Z., Naim, M., Raspat, J., Baker, D. J., Beck, J. E. *Adapting to When Students Game an Intelligent Tutoring System*. Proceedings of the 8th International Conference on Intelligent Tutoring Systems, 392-401.

(Baker et al., 2004) Baker, R.S., Corbett, A.T., Koedinger, K.R. (2004) *Detecting Student Misuse of Intelligent Tutoring Systems*. Proceedings of the 7th International Conference on Intelligent Tutoring Systems, 531-540

(Brace et Nejdl, 2004) Brace, J. et Nejdl W. (2004). "Ontologies and Metadata for eLearning". In Handbook on Ontologies 2004, S. Staab and R. Studer (Eds.). Springer-Verlag, 555-573.

(Brusilovsky et Peylo, 2003) Brusilovsky, P. et Peylo, C. (2003) *Adaptive and intelligent Web-based educational systems*. International Journal of Artificial Intelligence in Education 13 (2-4), Special Issue on Adaptive and Intelligent Web-based Educational Systems, 159-172.

(Bloom, Englehart, Furst, Hill et Krathwohl, 1956) Bloom, B. S., Englehart, M. D., Furst, E. J., Hill, W. H. and Krathwohl, D. R. (1956). *Taxonomy of educational objectives. Handbook I: Cognitive domain*. New York, NY : McKay.

(Carr & Goldstein, 1977) Carr, B., Goldstein, I. (1977). *Overlays: A theory of modelling for computer aided instruction*, Technical Report AI Memo 406, MIT, Cambridge, MA.

(Conati et al., 2002) Conati, C., Gertner, A., Vanlehn, K. (2002). *Using Bayesian networks to manage uncertainty in student modeling*. Journal of User Modeling and User-Adapted Interaction, volume 12, pages 371-417.

(de Gruijter, 1980) de Gruijter, D.N.M (1980). *A two-stage testing procedure*. Psychometrics for educational debates. New York: John Wiley and Sons.

(Desmarais & Pu, 2005) Desmarais M.C., Pu X (2005). *A Bayesian student model without hidden nodes and its comparison with Item Response Theory*. International Journal of Artificial Intelligence in Education, volume 15.

(Dufresne, 2001) Dufresne, A. *ExploraGraph : Improving interfaces to improve adaptive support*. Proceedings of AIED (Artificial Intelligence for EDucation) 2001, San Antonio, Texas.

(Gagné, 1996) Gagné, R. (1996). *Principles of Instructional Design*. Fort Worth, TX: HBJ College Publishers.

(Kashima, 2000) Kashima, Y. (2000). *Conceptions of culture and person for psychology*. Journal of Cross-cultural Psychology, pages 14-32.

(Linn et Gronlund, 1995) Linn, R. L. et Gronlund, N. E. (1995). *Measurement and assessment in teaching*. Upper Saddle River, NJ : Prentice-Hall

(Magis, 2007) Magis,D. (2007). *Influence, information and item response theory in discrete data analysis*. Thèse de doctorat. Université de Liège.

(Minsky, 1975) Minsky, M. (1975). A framework for representing knowledge. In P. Winston (Ed.), *The Psychology of Computer Vision*. New York: McGraw-Hill.

(Myslevy et Gitomer, 1996) Myslevy, R., Gitomer, D (1996). *The role of probability-based inference in an intelligent tutoring system*. User-Modeling and User- Adapted Interaction, volume 5, pages 253-282.

(Nkambou, Frasson et Gauthier, 1996) Nkambou, R., Frasson, C., et Gauthier, G. (1996). *CREAM-Tools: An Authoring Environment for Curriculum and Course Building in an Intelligent Tutoring System*. Paper presented at the Proceedings of the International Conference on Computer Aided Learning and Instruction in Science Engineering, San Sebastian, Spain.

(Nkambou et Tchetagni, 2004) R. Nkambou, J. P. M. Tchetagni (2004). *Diagnosing student errors in e-learning environment using MPE theory*. Web-based Education Journal – page 416-158

(Pearl et Russell, 2003) Pearl, J., et Russell, S. (2003). Bayesian Networks. In M. Arbib, A. (Ed.), *Handbook of BrainTheory and Neural Networks* (pp. 157-160). Cambridge, MA: MIT Press.

(Raïche, 2000) Raïche, G. (2000). *La distribution d'échantillonnage de l'estimateur du niveau d'habileté en testing adaptatif en fonction de deux règles d'arrêt: selon l'erreur-type et selon le nombre d'items administrés*. Thèse de doctorat inédite. Montréal: Université de Montréal.

(Resnick et al., 1994) Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J., (1994). "GroupLens: An Open Architecture for Collaborative Filtering of Netnews". In

(Resnick et al., 1994) Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J., (1994). "*GroupLens: An Open Architecture for Collaborative Filtering of Netnews*". In Proceedings of the Conference on Computer Supported Cooperative Work, Chapel Hill, NC, pp. 175-186.

(Sowa, 2000) Sowa, J. (2000). *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Toronto: Brooks/Cole.

(Tchetagni, 2005) J. P. M. Tchetagni (2005) *Une approche pro-pédagogique du diagnostic cognitif dans les sti: conception, formalisation et implémentation*. Thèse présentée par Joséphine-muriel Pelle-Tchétagani, comme exigence partielle du doctorat en informatique cognitive. 2005.

(Wainer, 2000) Wainer, H. (2000). *Computerized adaptive testing: A primer*. Mahwah, NJ: Lawrence Erlbaum Associates.

(Wenger, 1987) Wenger, E. (1987). *Artificial intelligence and tutoring systems: Computational and cognitive approaches to the communication of knowledge*: Morgan Kaufman Press.

(Weiss, 1982) Weiss D., (1982) Improving *measurement and efficiency with adaptive testing quality*. Applied Psychological Measurement, page 473-492

(Woolf et al, 2005) Woolf, B., Jonsson, A., Johns, J., Mehranian, H., Arroyo, I., Barto, A., Fisher, D., and Mahadevan, S (2005). *Evaluating the Feasibility of Learning Student Models from Data*. American Association for Artificial Intelligence Workshop on Educational Data Mining.

Articles et actes de colloques

Sodoké, K., Riopel, M., Raîche, G., Nkambou, R. et Lesage, M. *Extending Moodle functionalities to adaptive testing framework*. Article accepté pour la conférence internationale E-Learn 2007. Québec City, Canada.

Sodoké, K., Raîche, G. et NKambou, R. (2007). *La plate-forme d'évaluation adaptative des apprentissages: PersonFit*. Article soumis pour la conférence internationale ELAH (Environnement Informatique pour l'Apprentissage Humain) 2007. Lausanne, Suisse.

Sodoké, K., Raîche, G. et NKambou, R. (2007). *The adaptive and intelligent testing framework: PersonFit*. Article accepté pour la conférence internationale ICALT (IEEE International Conference on Advanced Learning Technologies), 2007. Niigata, Japon.

Sodoké, K., Raîche, G. et NKambou, R. (2006, soumis). *From adaptive testing framework to Intelligent Assessment Systems*. Communication soumise a la 7ème International Conference on Information Technology based Education and Training. Sydney, Australie.

Sodoké, K., Raîche, G. et NKambou (2006, soumis). *Système de test adaptatif à un agent de recommandation de questions*. Actes de du colloque de l'Institut du conseil national de recherches pour les technologies didactiques. Genève, Suisse.

Sodoké, K., Raîche, G. et NKambou (2006). *D'un système de test adaptatif à un agent de recommandation de question*. Article accepté et publié comme acte pour la conférence international TICE (Technologies de l'Information et de la Communication en Éducation) 2006. Toulouse, France.

Communications présentées dans un colloque

Sodoké, K. et Riopel M. (2006). *La complémentarité des réseaux bayésiens et des modélisations issues de la théorie de la réponse à l'item dans les tests adaptatifs*. Communication présentée au 52ème congrès de ACFAS (Association Francophone pour le Savoir). Trois-Rivières, Québec - Canada.

Sodoké, K. et Riopel M. (2006). *La mise à l'essai d'une plate-forme d'évaluation sur mesure : webQuiz*. Communication présentée au 52ème congrès de ACFAS (Association Francophone pour le Savoir). Montréal, ACFAS – McGill University.

Sodoké, K., Raïche, G. et NKambou, R. (2006). *D'un système de test adaptatif à un agent de recommandation de questions*. Communication présentée au congrès annuel du CIRTÀ (Centre Interuniversitaire de Recherche sur le Téléapprentissage). Montréal, CIRTÀ – ACÉD et de AMTEC 2006.

Sodoké, K., Raïche, G., Riopel M et NKambou, R. (2006). *Conception et développement de tests adaptatifs par ordinateur avec intégration dans Moodle*. Communication proposée au congrès annuel de l' ADMEE (Association pour le Développement des Méthodologies d'Évaluation en Éducation). Trois-Rivières, Québec - Canada.